

# Vensim<sup>®</sup>6 ユーザーガイド

日本未来研究センター  
システムダイナミックスグループ

樽本祐助 訳





# はしがき

日本未来研究センターは、2002年4月に設立されたNPO法人で、以下の3つのミッション（使命）を遂行し、日本および世界の未来の持続可能な発展（Sustainable Development）に貢献すべく、微力ながら10年以上にわたり活動に取り組んできました。

- システムダイナミックスの普及
- 未来ビジョンが交換・学習できる交流の場（フォーラム）の提供
- 持続可能な地球モデル村および未来博物館の建設・運営

すなわち、未来をコンピュータ・シミュレーションの手法を用いてデザインし、コンピュータではデザインできない場合には私たちの頭脳を用いてデザインし、それらに立脚しつつ持続可能な地球村、社会を建設してゆきたいというのが私たちのミッションです。

その最初の普及ミッションであるシステムダイナミックス（System Dynamics）とは、ストック、フロー、変数、矢印の4つのアイコンを組み合わせることにより、企業の経営戦略、政府・自治体の公共政策、地球環境問題等々多岐にわたる複雑なシステムをお絵かき感覚でモデリングし、その構造をシミュレーション分析する手法で、1950年代にMIT（マサチューセッツ工科大学）のジェイ・フォレスト教授によって開発されたものです。4つのアイコンを組み合わせでお絵かきをするといった簡単なモデリング手法のために、米国ではK-12（初等・中等）教育にもシステムダイナミックスが普及しています。

私たちは、システムダイナミックスの代表的ソフトである Vensim の国内販売をさせていただく機会をえ、これまでその日本語化にも取り組んできました。今回、同センターのシステムダイナミックス研究員である樽本祐助さんの尽力により、長年の念願であった Vensim 6 User's Guide の日本語訳が完成し、皆さんにお届けすることができるようになりました。これを機会にシステムダイナミックスによるシステム構造分析、未来システム思考が広く普及し、日本の社会がより持続可能な社会に進化してゆくことを念願しています。

NPO 法人日本未来研究センター

理事長 山口 薫, Ph.D.

2013年4月



# 序

システムダイナミクスについての関心から、Vensim を利用し始めて数年になります。その利用のなかで、試行錯誤でモデルを作成してきたが、まだ十分に活用できているとは思えません。

また、わが国でのシステムダイナミクスの活用も諸外国に比べて少なく、具体的なモデルの作成方法についての情報もすくないように思います。また Vensim に関する資料も日本語で参考になるものはあまりありません。

そこで自分自身の勉強もかねて、Vensim User's Guide を翻訳したのが本書です。最新の Vensim 環境で、検証しながら作業を行っています。こうした作業のなかで、記載内容も見直しています。

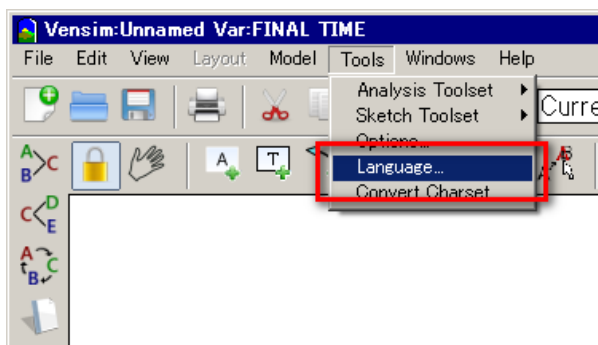
一方で、内容の理解が十分でないために、わかりにくい文章も残されていると思っています。

しかし、こうした資料として公表することで、より良い資料として改善していけたらと考えてます。また、この資料がシステムダイナミクスの利活用推進や、Vensim を用いたシミュレーションモデルの開発の一助になることも願っています。

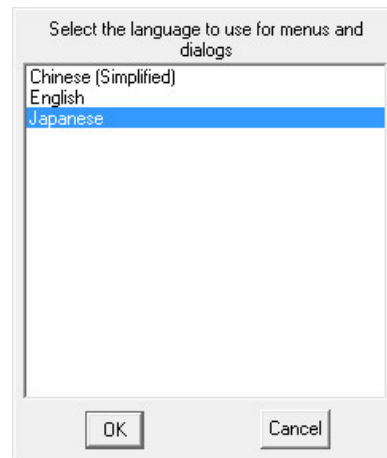
Vensim では、メニューの日本語化が進められています。


初期設定では、英語メニューになっているので、日本語メニューにするには以下の方法をとります。

1. Vensim を起動します。
2. [ Tools ] メニューにある [ Language ] をクリックします。



3. ダイアログに表示される [ Japanese ] をクリックし、[ OK ] をクリックします。

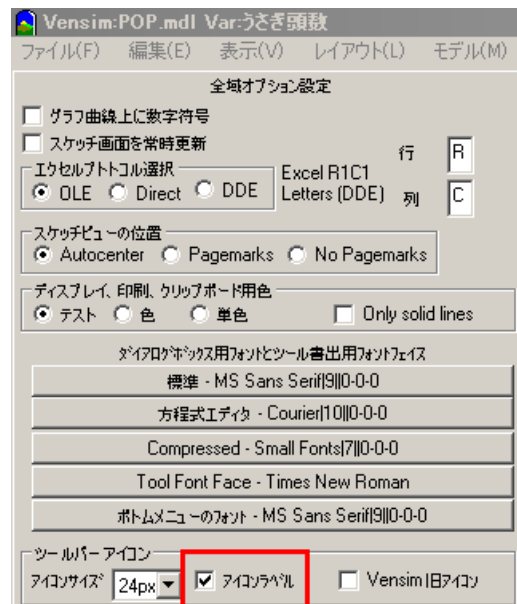


4. Vensim を閉じるために、Vensim 画面の右上にある  をクリックします。
5. Vensim を改めて起動すると、日本語メニューとなります。

さらに、アイコンの下にラベルが表示されます。この日本語への対応が完全ではないので、以下の方法で表示を消して、ラベルの上にマウスを動かすことで表示されるポップアップによりツールを確認する方が便利です。そのためには、つぎのように操作します。

#### Windows PLE 版

1. [ オプション ] メニューにある [ オプション ] をクリックします。
2. ツールバーアイコンにある [ アイコンラベル ] のチェックを外します。



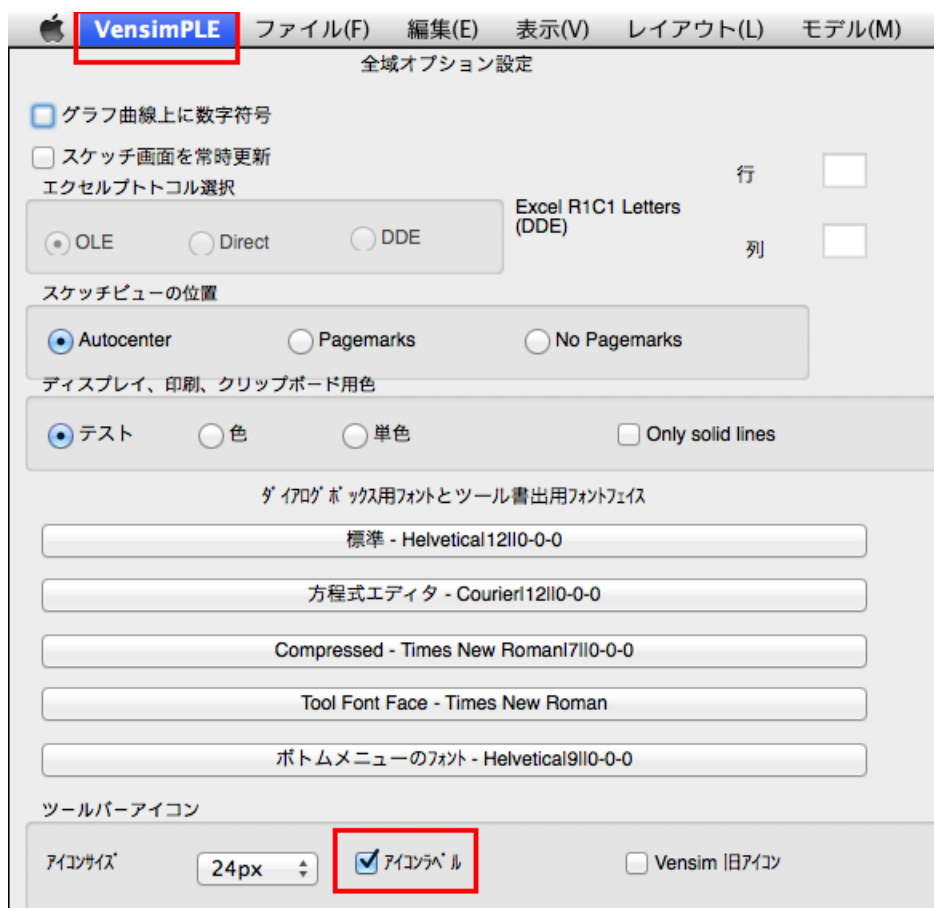
#### Windows Pro 及び DSS 版

1. [ ツール ] メニューにある [ オプション ] をクリックします。
2. [ ツールバーアイコン ] タブを選択し、[ アイコンラベル ] のチェックを外します。



### Mac PLE 版

1. [ VensimPLE ] メニューにある [ Preferences ] をクリックします。
2. ツールバーアイコンにある [ アイコンラベル ] のチェックを外します。



### Mac Pro 及び DSS 版

1. [ VensimDSS ] メニューにある [ Preferences ] をクリックします。
2. [ ツールバーアイコン ] タブを選択し、[ アイコンラベル ] のチェックを外します。



樽本祐助, 博士 (農学)

2013 年 4 月

# 目次

はしがき	i
序	iii
第 1 章    イントロダクション	1
1.1    Vensim について	1
1.2    このユーザーズガイドについて	1
1.2.1    ガイドの構成	2
1.2.2    表記法	3
1.3    重要なお知らせ	4
1.3.1    ディレクトリ	4
1.3.2    プログラムの概観	4
1.3.3    マウス	4
1.3.4    タブダイアログボックス	5
1.4    Vensim のインストール	5
1.4.1    インストールプログラムの実行	7
1.4.2    ライセンス契約	7
1.4.3    登録コード	7
1.4.4    インストールディレクトリ	8
1.4.5    その他の設定	8
1.5    その他の情報元	9
第 2 章    Vensim のユーザインターフェース	11
2.1    主な特徴	11
2.2    タイトルバー	12
2.3    メニュー	12
2.4    ツールバー	13
2.5    ウィンドウの種類	13
2.5.1    ウィンドウ間の移動	14
2.5.2    同じ種類のウィンドウ間の移動	14
2.5.3    モデルウィンドウ	14
2.5.4    スケッチツール	15
2.5.5    ステータスバー	16
2.5.6    シミュレーション	16
2.6    出力ウィンドウ	16
2.6.1    分析ツール	17
2.6.2    分析ツールの出力	18
2.7    制御パネル	19
2.8    下添え字コントロール	20
第 3 章    実際の操作	21
3.1    Vensim によるモデリング	21

3.1.1	労働力と在庫の例	21
3.2	Vensim の開始	21
3.2.1	モデルを開く	22
3.3	構造の確認	22
3.4	モデルのシミュレーション	24
3.5	振る舞いの検討	25
3.6	原因追跡	27
3.7	個別のシミュレーション実験	30
3.8	カスタムグラフの作成	32
3.9	カスタム表の作成	34
3.10	要約	36
第 4 章	因果ループ図	37
4.1	Vensim モデル	37
4.1.1	スケッチを描く	37
4.1.2	マウスのヒント	38
4.2	因果ループ図の作成	38
4.2.1	プロジェクト・モデル (project.mdl)	38
4.3	図形の修正	42
4.3.1	スケッチのレイアウト	42
4.3.2	スケッチオプション	43
4.3.3	コメントと画像の追加	45
4.3.4	モデルの改良	46
4.4	スケッチの印刷	47
4.5	分析ツール	48
4.6	樹形図による因果分析	48
4.6.1	分析ツールオプション (Vensim PLE と PLE Plus は対応しません)	50
第 5 章	ストックおよびフロー図	55
5.1	作図 (customer.mdl)	55
5.1.1	ストックの入力	55
5.1.2	フローの作成	56
5.1.3	曲がるフローパイプ	57
5.1.4	補助変数と矢印の追加	57
5.1.5	より詳しい構造	58
5.2	図のカスタマイズ	59
5.2.1	スケッチオプション	60
5.2.2	変数の形状	60
5.2.3	デフォルトのスケッチオプション (PLE と PLE Plus は対応しません)	61
5.2.4	スケッチ上の要素を隠す (PLE と PLE Plus は対応しません)	62
5.2.5	画面表示の変更	63
5.3	スケッチツールの修正 (PLE と PLE Plus は対応しません)	63
5.3.1	スケッチツールセットの保存	64
第 6 章	シミュレーションモデルの構築	65
6.1	うさぎ頭数モデル	65
6.2	Vensim の表記法	65
6.3	うさぎ頭数モデルの作成 (pop.mdl)	66
6.4	方程式の入力	67
6.5	構文と単位のチェック	71
6.5.1	単位の同意性	71
6.5.2	モデルのシミュレーション	72



6.6	モデルの分析	72
6.7	シミュレーションの比較	73
6.7.1	指数的な増加	73
6.7.2	シミュレーション実験	73
6.7.3	シミュレーションのセットアップによる方法	74
6.7.4	直接原因グラフ	75
6.7.5	シミュレーションの比較	76
6.7.6	指数的な減少	76
6.7.7	入出力オブジェクト	77
第 7 章	関数とシミュレーションエラー	79
7.1	価格と需要	79
7.2	関数の入力	80
7.2.1	IF THEN ELSE 関数	83
7.2.2	SMOOTH 関数	84
7.2.3	STEP 関数	85
7.2.4	初期条件	85
7.3	モデル方程式	86
7.4	シミュレーションエラー	87
7.4.1	時間がゼロのエラー	88
7.4.2	シミュレーション中のエラー	89
7.4.3	不可解な変化	90
7.4.4	エラーの統合	92
7.5	モデルの修正	92
7.5.1	XIDZ	92
第 8 章	表関数の作成	93
8.1	うさぎ増加の限界	93
8.2	表関数の標準化	94
8.2.1	モデルの作成 (rabbit.mdl)	95
8.2.2	方程式の入力	96
8.3	表関数の作成と標準化	97
8.3.1	値の編集	98
8.4	構文と単位のチェック	99
8.5	モデルのシミュレーション	99
8.6	モデルの分析	99
8.7	標準化された変数の分離	100
8.7.1	シミュレーション	102
8.7.2	表関数の変更	102
8.8	表関数の変数	103
第 9 章	複数の表示画面	105
9.1	表示画面の働き	105
9.1.1	グラフィックモデルと変数	105
9.2	顧客の増加モデル	106
9.2.1	顧客の増加モデルの作成 (cust1.mdl)	106
9.2.2	方程式の入力	107
9.2.3	構文と単位のチェック	107
9.2.4	モデルのシミュレーション	108
9.2.5	モデルの分析	108
9.2.6	表示画面の名称とモデルの保存	108
9.3	表示画面「能力」の追加 (cust2.mdl)	109

9.3.1	代行変数ツール	110
9.3.2	方程式の追加	111
9.3.3	方程式の変更	111
9.4	表示画面「販売収入」の追加	112
9.5	モデル変数ツール	112
9.5.1	方程式の追加	113
9.5.2	モデルのシミュレーション	113
9.5.3	モデルの分析	113
9.5.4	モデルの保存	114
9.6	詳細な能力モデル (cust3.mdl)	114
9.6.1	コピーと貼り付け	115
9.6.2	表示画面「能力」の方程式	117
9.6.3	単位同義語	118
9.7	モデルのシミュレーションと分析	118
9.8	生産能力の投資戦略	119
第 10 章	出力のカスタマイズ	121
10.1	分析ツールからの出力	121
10.1.1	樹形図	121
10.1.2	グラフと直接原因グラフ	122
10.1.3	細い線 (カラー)	122
10.1.4	太い線 (カラー)	123
10.1.5	線の種類 (カラー)	124
10.1.6	点のみのグラフ	125
10.2	マーカーのついた線	125
10.3	カスタムグラフ	127
10.4	カスタム表	129
第 11 章	ゲーム	133
11.1	ゲームとは?	133
11.1.1	不動産ゲーム (houses.mdl)	133
11.1.2	ゲームの実行	137
11.1.3	ゲームの前進	137
11.1.4	ゲームのさかのぼり	139
第 12 章	入出力コントロール	141
12.1	口こみによる販売数	141
12.1.1	wom1.mdl の方程式	142
12.2	出力コントロール	143
12.3	入力コントロール	144
12.3.1	宣伝費のスライダ	145
12.3.2	製品寿命のスライダ	145
12.3.3	他の変更可能な定数	145
12.3.4	整列	146
12.3.5	モデルのシミュレーション	146
12.4	ゲームコントロール	147
12.4.1	新しい表示画面への貼り付け	147
12.4.2	ゲームの実行	147
12.5	モデルの公表	148
12.5.1	ゲームの間隔	148
12.5.2	コメントとナビゲーション	149
12.5.3	ナビゲーション	149

12.5.4	稼働テスト	151
12.5.5	パッケージの公表	151
第 13 章	統合シミュレーション	153
13.1	軍備競争モデル	153
13.1.1	図の新しい装飾	154
13.1.2	長い接続	154
13.1.3	収納枠	154
13.1.4	最小値と最大値と増分	155
13.1.5	モデル方程式	155
13.2	構造と振る舞いとスピード	157
13.2.1	振る舞いの表示	157
13.2.2	シミュレーションの作成	158
13.2.3	スライダ値の設定	159
13.2.4	矢印キーの利用	160
13.2.5	表関数の変更	160
13.2.6	ツールバーでの変更 (PLE と PLE Plus は対応しません)	161
13.3	シミュレーション実験	162
13.3.1	実行結果の読み込みと取り除き	163
13.4	フィードバックループの遮断	163
13.4.1	競争の中止	164
13.4.2	振る舞いの優先	165
13.5	制御パネルの作成	165
13.6	下添え字と統合シミュレーション (DSS と Professional 版のみ対応)	167
第 14 章	リアリティチェック	169
14.1	モデルと現実性	169
14.1.1	専門家の役割	170
14.2	リアリティチェック方程式の定義	170
14.2.1	テスト入力	171
14.2.2	制約	171
14.2.3	結果における動的なテスト	173
14.2.4	:CROSSING:	173
14.2.5	:AT LEAST ONCE:	173
14.2.6	空の条件	174
14.2.7	結果における未知のテスト	174
14.3	シミュレーションとリアリティチェック	174
14.3.1	実行中の制約チェック	174
14.3.2	受動的な制約チェック	175
14.3.3	エラーのリポート	175
14.4	リアリティチェック方程式の入力	175
14.4.1	方程式エディタ	176
14.5	リアリティチェックの実行	176
14.5.1	リアリティチェックツール	178
14.5.2	リアリティチェックの結果	178
14.5.3	シミュレーション結果の検討	178
14.6	イースト菌の成長における現実性チェック	179
14.6.1	テスト入力と制約方程式	179
14.6.2	初期モデル	181
14.6.3	温度と分裂そして終結	184
14.6.4	水と砂糖から影響を受ける分裂	185
14.6.5	分裂により影響を受ける水と砂糖	186

14.6.6	結論	187
第 15 章	感度テスト	189
15.1	モンテカルロシミュレーション	189
15.2	市場成長モデル (sales.mdl)	189
15.2.1	Sales.mdl の方程式	190
15.2.2	初期設定のシミュレーション	190
15.3	複数のパラメータによる不確実性	191
15.3.1	感度分析の管理パラメータ	191
15.3.2	無作為の一樣分布	192
15.3.3	無作為の正規分布	193
15.4	保存リスト	194
15.5	感度分析シミュレーション	195
15.6	感度分析の結果出力	195
15.6.1	信頼幅	195
15.6.2	平均値 (Vensim PLE Plus は対応しません)	196
15.6.3	グラフ目盛りの拡大	196
15.6.4	個別シミュレーションの表示	197
15.7	感度分析のヒストグラム表示	197
15.7.1	デフォルトの分析ツールセットの変更	198
15.7.2	統計ツール	199
第 16 章	モデルにおけるデータ利用	201
16.1	データ利用のタイプ	201
16.2	データを使ったモデル操作 (cfc.mdl)	201
16.2.1	cfc.mdl 方程式	202
16.3	表計算ソフトからのデータ読み込み (PLE は対応しません)	202
16.3.1	シミュレーション	203
16.4	読み込まれたデータ変数	204
16.5	テキスト形成のデータ読み込み (.dat)	206
16.5.1	シミュレーション	206
16.5.2	ツールバー	207
16.5.3	シミュレーション制御 (PLE と PLE Plus は対応しません)	207
16.5.4	結果	208
16.6	表計算データの読み込み (PLE は対応しません)	209
16.6.1	シミュレーション	212
第 17 章	下添え字	213
17.1	下添え字とは	213
17.2	簡単なプロジェクトモデル (proj1.mdl)	214
17.2.1	proj1.mdl 方程式	214
17.3	複数の仕事を持つプロジェクト (proj2.mdl)	215
17.3.1	下添え字範囲	215
17.3.2	下添え字コントロール	215
17.3.3	変数への下添え字の追加	216
17.3.4	定数の扱い	217
17.3.5	ベクトル関数	217
17.3.6	方程式編集ダイアログによる下添え字の作成	218
17.3.7	下添え字の要約	218
17.3.8	シミュレーションと分析	219
17.4	下添え字モデルの応用 (proj3.mdl)	220
17.4.1	下添え字範囲	220

17.4.2	下添え字の写像	221
17.4.3	下添え字を持つ変数の多元方程式	222
17.4.4	論理関数と制御	224
17.4.5	VMIN 関数	224
17.4.6	シミュレーションと分析	225
第 18 章	最適化	227
18.1	最適化の概要	227
18.2	モデルの補正 (electric.mdl)	227
18.2.1	Electric.mdl 方程式	228
18.2.2	データの比較	229
18.2.3	ペイオフの設定	230
18.2.4	最適化パラメーターの設定	231
18.2.5	信頼区間の計算	233
18.2.6	モデルの修正	234
18.3	方策の最適化 (stock.mdl)	234
18.3.1	在庫モデル	234
18.3.2	手動の最適化	238
18.3.3	ペイオフの設定	239
18.3.4	最適化パラメータの設定	240
第 19 章	モデルの配布	243
19.1	Vensim Model Reader	243
19.1.1	モデルの下準備	244
19.1.2	パッケージモデルの出版	245
19.1.3	パッケージの出版	245
19.1.4	バイナリ形式	247
19.2	パッケージモデルを開く	247
19.3	Vensim PLE によるモデルの利用	248
19.4	E メールとウェブ利用上の注意	249
第 20 章	参照モード	251
20.1	参照モードによる概念化	251
20.1.1	ループの作成	254
20.1.2	シミュレーション	256
20.1.3	メンタルの統合	258
20.1.4	参照モードの管理	259



第 1 章

イントロダクション

1.1 Vensim について

Vensim は、ダイナミックシステムのモデルを概念化し、文書化し、シミュレーションし、分析し、最適化できる視覚的なモデリングツールです。Vensim により、因果ループあるいはストックおよびフロー図からシミュレーションモデルを簡便に構築することができます。

言葉を矢印で結び付けることによって、システム変数の関係は因果関係として扱われます。この情報はシミュレーションモデルを完成させるための方程式で使われます。シミュレーションモデルの作成プロセスにおいて、変数の利用状況を分析することができます。シミュレーション可能なモデルに対して、その振る舞いを詳しく解析することができます。

1.2 このユーザースガイドについて

このユーザースガイドは、Vensim の主な特徴を説明します。Vensim に付属する既存のモデルを実際に操作することによって、Vensim の紹介、因果ループ図の作成、ストックおよびフロー図の作成、シミュレーションモデルの作成を行います。

Vensim の応用機能となる感度分析や下添え字（配列）、最適化などは、既存のシミュレーションモデルを使用することで学習を進めます。このガイドにあるモデルはすべて（3 章、14 章、18 章は除く）自習できるように操作可能な形式で付属しています。

付属のモデルは英語です。

このガイドはすべての Vensim のラインアップで共通です。特定のラインアップにはない機能は明記します。下記の表は、各章がどのラインアップに対応しているのかを示しています。

章	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
PLE																				
PLE Plus																				
Standard																				
Professional																				
DSS																				

Vensim の新規ユーザには、このガイドは始めから終わりまで通読できるように構成されています。経験があるユーザは、3 章、7 章および 13 章を勧めます。そこでは新たな

機能が紹介されています。

### 1.2.1 ガイドの構成

このガイドは、3 部構成です。最初の 1 章から 3 章は、このソフトウェアの紹介部分です。4 章から 10 章は、Vensim によるモデル作成の方法を説明します。つまり、作図、方程式の追加、モデルのシミュレーションと分析、結果の表示方法が示されます。11 章から 18 章は、Vensim の応用機能を既存のモデルを使って説明します。

Vensim では、ストックは Level【レベル】、フローは Rate【レート】という用語も使われています。この日本語マニュアルではストックとフローという用語で統一します。

- 1 章 このガイドと Vensim を紹介します。また Vensim をインストールするための説明があります。
- 2 章 Vensim のユーザインターフェースを紹介します。本章は、スケッチツール、分析ツールおよびコントロールウィンドウといった Vensim の機能を紹介します。
- 3 章 既存のモデルを用いたシミュレーション分析を体験します。
- 4 章 因果ループ図の作成方法を紹介し、また、分析ツールを使用した因果ループ図の構造分析を示します。
- 5 章 ストックおよびフロー図を作成します。
- 6 章 人口増加のシミュレーションモデルを順序立てて作成します。ここでは、Vensim によるモデルの作成、シミュレーション、分析の方法を習得します。
- 7 章 モデルにおける関数の利用方法を示し、さらにシミュレーションにおいて生じるエラーを検知したり、修正する方法を紹介し、
- 8 章 表関数の作成と使用方法を記述します。表関数は、関係を示すグラフによって、入力と出力を関連づける機能です。これらはテーブル関数とも呼ばれます。
- 9 章 モデルを部門に分割する方法を示します。複数の表示画面を持つモデルを開発します。
- 10 章 分析ツールにより出力されるグラフをカスタマイズする方法を示します。さらにカスタムグラフおよびテーブルエディタによって、複数の変数を用いた出力方法を説明します。
- 11 章 モデルをゲームとして、あるいはフライトシミュレータとして使用する方法を紹介します。それは、時間を段階的に区切ってシミュレーションするとともに、ある時点での意思決定をシミュレーションに反映させます。Vensim PLE は本章に対応しません。
- 12 章 入出力コントロールやナビゲーションリンクの使用法を示します。これらはモデルのプレゼンテーションに利用する機能です。
- 13 章 統合シミュレーションがどのように用いられるのかを示します。統合シミュレーションは、Vensim のバージョン 5 からの新機能です。
- 14 章 リアリティチェックの特徴を示し、モデルの妥当性を検討します。
- 15 章 モンテカルロによる感度分析の例を示します。値が確定していない変数に対して、何度かのシミュレーションによる感度分析によって、その変数の値を与えます。Vensim PLE は本章に対応しません。
- 16 章 モデルの中でデータを使用する方法について紹介します。データ変数は、外生の時系列データであり、モデルの振る舞いをもたらします。本章は、テキストファイル



やスプレッドシートからのデータ読み込みを紹介します。一部の機能で、Vensim PLE は対応しません。

17 章 下添え字（配列）の単純な例と、より複雑な例を紹介します。本章は Vensim Professional と DSS のみ対応しています。

18 章 2 つのタイプのモデル最適化を扱います。最初のタイプは外部データにモデル出力を適合させるモデル定数の調整（キャリブレーション）です。もう一つは、方針の最適化であり、収益や支払いといった目的関数の最大化や最小化のためのモデルパラメータを見つけます。本章は Vensim Professional と DSS のみ対応しています。

19 章 モデルを他の人が利用するための方法やヒントを示します。Vensim Model Reader や開発されたラインナップとは異なる Vensim での利用方法を示します。

20 章 参照モードを利用するための方法を紹介합니다。これはフィードバックモデルを概念化する方法であり、メンタルを統合する手段でもあります。

### 1.2.2 表記法

Vensim で用いられる多くの要素を区別するため、このガイドをはじめとしたマニュアルは、以下の表記法をとっています。


- 保存されたファイル名やその拡張子は、イタリック体（例えば *project.mdl*）で示されます。データセットの名前は *.vdf* という拡張子のないイタリック体（例えば *baserun*【基礎実行】）で示されます。
- Vensim モデル中の変数および方程式の名前はイタリック体等幅フォント（例えば *Population*【人口】）で示されます。
- Vensim のメニュー項目、コントロール、ボタン、ツール、ツールバーおよびダイアログボックスの名前はすべて大文字を使用します（例えば Control Panel）、特に選択や行動がともなう場合は、太字を使用する、（例えば、Simulate ボタンを押す）。
- 演習で行う操作は、三角形の弾丸マークを使用します。

データセットとは、シミュレーションの結果が保存されたファイルのことです。

例えば以下のように表記します。

➤ Open Model ボタンをクリックして、モデル *wfinv.mdl* を選んでください。

英語のマニュアルやサンプルファイルを見るには、上記の表記法を知っておくとよい。この日本語マニュアルでは、上記に加えて以下のような表記法を用います。

- 英文の訳には【】を用います。例えば、Pen【ペン】。
- メニュー操作では、[ ] と を用います。例えば、[ ファイル ] メニュー [ 保存 ] を選択します。
- ツールについては、ツール名とそのアイコンを用います。例えば、スケッチ移動 。
- 入力項目は、「」を用います。例えば、「在庫」と入力します。
- キーボード操作については、キーを囲みます。例えば、**Shift** キーを押します。

## 1.3 重要なお知らせ

### 1.3.1 ディレクトリ

このマニュアルで扱われるすべてのモデルは、インストール時に設定したサンプルモデルの保存場所である *guide* サブディレクトリにあります。比較的新しい OS では、*C:\Users\Public\Vensim\models* であり、場合によっては *C:\Program Files\Vensim\models* になることもあります。

通常のモデル作成では、上記の Vensim のサブディレクトリ以外の場所にモデルを保存することを勧めます。つまりこのガイドにそった作業だけを、*guide* のサブディレクトリに保存します。

### 1.3.2 プログラムの概観

Vensim PLE、PLE Plus、Standard、Professional および DSS の概観（メニューやボタン）には若干の違いがあります。さらにツールセットは変更が可能です。英文マニュアルでの概観は基本的に Vensim DSS とデフォルトのツールセットを使用しています。もし利用されている Vensim と多少異なる概観であっても、それは問題ではありません。

この日本語マニュアルでは Professional を主に利用しています。

### 1.3.3 マウス

Windows コンピューターは左右のマウスボタンを持っています。Vensim は、下記のように左右のボタンを利用します。

マックではボタンが 1 つしかありません。

マックでは、**Ctrl** や **⌘** をクリックの際に併用します。

#### 左ボタン

左ボタンはメニューの選択やボタンのクリック、グラフやスケッチオブジェクトのドラッグといった Vensim の大部分のオペレーションで使用します。ガイドで、左ボタンや右ボタンなどの説明がなく、マウスをクリックすると記載されている場合は、左ボタンを使います。

マックでは、ボタンをクリックします。

#### 右ボタン

右ボタンはスケッチツール、分析ツールおよびスケッチオブジェクトのオプションを設定するために使用します。またスケッチのポジションやズームに利用します。

右ボタンをスケッチ上でドラッグするとスクロールします。これは左マウスをクリックしたまま **Ctrl** キーを押してドラッグすることと同じです。ズームを行うには、**Shift** キーを押したまま右ボタンをドラッグします。

マックでは、**Ctrl** + クリックか、もしくは **⌘** + クリックとします。

### ホイ尔マウス

ホイ尔マウスがある場合は、ウィンドウを上下にスクロールすることができます。水平スクロールには[Shift]キーを押したまま、ホイ尔を回転させます。[Ctrl]キーを押したまま、ホイ尔を回転させれば、スケッチが拡大もしくは縮小します。

#### 1.3.4 タブダイアログボックス

タブダイアログボックスはタブを持つ異なるフォルダに情報を分けることにより、管理を簡便にします。適切なタブをクリックすることによりフォルダを切り換えることができます。タブダイアログボックスは、シミュレーションコントロールや方程式エディタ、制御パネル（下記の例）で用いられます。



この画面では、[データセット]タブが選択され、2つのシミュレーションがあることを示しています。[変数][時間軸][目盛り]などは、タブをクリックすることで選択できます。

## 1.4 Vensim のインストール

Vensim をインストールするには、インストールプログラムの実行が必要です。これはCD もしくは、ウェブサイト <http://www.vensim.com> から入手します。

### CD によるインストール

Vensim CD は、すべてのラインアップのインストールプログラムが含まれています。CD のラベルにはバージョン番号があり、登録コードを持つラインアップだけをインストールすることができます。

### ダウンロードによるインストール

ウェブサイト <http://www.vensim.com> から Vensim をダウンロードすることができます。Vensim ライセンスは、1年間のバージョンアップのための保守と技術サポートを含みます。その後の保守とサポートを継続するには、メンテナンス料が必要になります。どのバージョンが利用可能であるかは、Vensim のウェブサイトを見ることでわかります。

また、[ヘルプ]メニュー [最新版]により、より新しいバージョンが利用可能かどうか確認できます。

Vensim をダウンロードするための直接リンクは <http://www.vensim.com/cgi-bin/download.exe> です。そこにアクセスするには、登録コードの入力が求められます。

登録コードにより、所有する製品のラインアップに加えて、契約開始日、メンテナンス契約の有無を確認します。

一旦登録コードを入力すれば、利用可能な Vensim のバージョンを選択することができます。もし必要とする特定のバージョンがなければ、最新のバージョンを選んでください。

教育目的のための PLE のダウンロードには、<http://www.vensim.com/freedownload.html> にアクセスします。

#### Windows へのインストール

Windows のインストーラーは数個のファイルに分割されています。これらのファイルの1番目は、ラインアップ (例えば Vensim DSS では *vendss32.exe*) に対応した名前を持っています。残りのファイルは *disk2.vip*、*disk3.vip* などの名前がつけられています。

ダウンロードする場合、必ず同じディレクトリにすべてのファイルを保存してください。またファイル名は決して変更しないでください。分割されたファイルの内容は、ダウンロードページに詳しく記載されています。最初のファイルは実際のプログラムです。*disk2.vip* にはサンプルモデルがあり、*disk3.vip* から *disk6.vip* はドキュメントです。Vensim DSS には、これらに加えて *disk7.vip* があり、Vensim DLL とサポートファイルがあります。

最新のプログラムをインストールする場合には、最初のファイルだけをダウンロードしてバージョンアップします。

**注意** ダウンロードした *.vip* を開いてはいけません。これらはインストールにおいて利用されますが、個別に開くことはできません。例え、1つのコンテンツをインストールする場合であっても、最初の (*.exe*) ファイルからの実行が必要です。

#### マックへのインストール

マック用のプログラムは、OSX Version 10.4 以上のもの (Intel) と、PowerPC 用のもの (PowerPC) の2種類があります。サンプルモデルやドキュメントは共通です。

ダウンロードしたファイルは、バイナリファイルで、マックのプログラムファイルとするために解凍します。

#### ブラウザベースのヘルプファイル

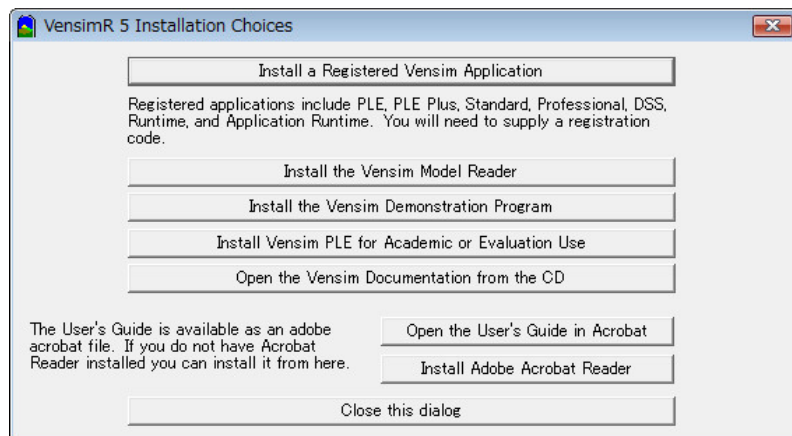
ヘルプファイルには、Vensim CD にある *.htm* ファイルが利用可能です。これらをハードディスクにコピーし、ブラウザで *vensim.htm* を開きます。さらに、<http://www.vensim.com/documentation/vensim.htm> で同じオンラインのヘルプファイルを見ることができます。

### 1.4.1 インストールプログラムの実行

#### CD によるインストール

CD を持っている場合、CD ドライブに挿入します。自動的にインストール選択の画面が表示されます。

もしこの画面が表示されない場合は、CD にある *setup.exe* をダブルクリックします。



インストール選択の画面から、インストールしたいプログラムを選択します。登録コードを持っている場合、[ Install a Registered Vensim Application ] をクリックし、登録コードを入力します。

#### ダウンロードによるインストール

Vensim のインストールプログラムをダウンロードした場合、ダウンロードした最初のファイル (例えば Vensim DSS では *vendss32.exe*) をダブルクリックします。

これらのファイルの場所は、ダウンロードの際に選択したディレクトリになります。

マックでは、.dmg 形式のため解凍して、アプリケーションとします。

### 1.4.2 ライセンス契約

Vensim をインストールする前に、ライセンス契約の条件に同意する必要があります。このライセンス契約は英文マニュアルの最後にも記載されています。ライセンス契約の条件に同意する場合は、インストールを続けます。ライセンス契約の条件に同意しなければ、支払った料金の払い戻しのためにソフトウェアを返却することもできます。

### 1.4.3 登録コード

商業利用のための Vensim DSS、Professional、Standard、PLE Plus および PLE には、登録コードが必要です。

教育もしくは評価目的のための PLE には登録コードは必要ありません。もし登録コードを持っていないのであれば、教育もしくは評価目的として PLE をインストールしてください。

登録コードは文字列、数字および線記号から構成されています。もしオンラインライセンスを購入するか、ライセンスを電子的に受けとった場合は、登録コードが電子メールによって送られてきます。そうでなければ、それはライセンス証明書、あるいは CD やそのケースにラベルがつけられています。登録コードは以下のようなものです。

Registration Code: ABCDE-FGHIJ-KLMNO-PRQS

Company: Ventana Systems, Inc.

Product: DSS

Serial#: 0

インストールの際、登録コードを正確に入力してください。線記号についてはスペースで代用することもできます。また会社名は、登録コードの下に記載された会社名と一致させます。電子メールによって登録コードを受け取ったならば、電子メールからそれをコピーし、それを貼り付けるのが簡単です。もし入力したコードが違っていれば、再入力が求められます。コードが受け取ったものと同一であるか再点検してください。

あなたの会社名が正しくない場合は Vensim に連絡してください。

#### 1.4.4 インストールディレクトリ

Vensim をインストールしたいディレクトリを選ぶことができます。Windows ではデフォルトでプログラムファイルのディレクトリ (通常 *C:\Program Files\Vensim*) になります。しかし Vensim は、どんな場所にもインストールすることができます。このマニュアルにおいてディレクトリを参照する場合は、Vensim をインストールしたディレクトリのサブディレクトリになります。

マックでは、アプリケーションディレクトリの中にプログラムが置かれます。

#### 1.4.5 その他の設定

Vensim Professional および DSS は *vensim.exe* という名前のプログラムをインストールします。したがって、Professional から DSS にアップグレードすれば、Professional が DSS に取って代わります。

一方で、Vensim PLE および PLE Plus は、異なる名前でインストールされます。PLE の実行ファイルは *venple.exe* であり、PLE Plus は *venplep.exe* となります。したがって、同じディレクトリに別のラインアップを矛盾なくインストールすることができます。

あるラインアップでは必要とされないサポートファイルもありますが、すべてのラインアップでインストールされるサポートファイルは同一です。さらに Vensim は、ファイルサイズの小さい設定ファイルを作成します。vensim.ini (Professional と DSS) venple.ini (PLE) および venplep.ini (PLE Plus) というファイルです。

このように設定ファイル名も違うため、同じディレクトリに複数の Vensim ラインアップをインストールすることができます。

## 1.5 その他の情報元

Vensim による作業では、簡単に有益な情報を利用できます。Vensim ウィンドウのボタンの名前や操作については、参照カードでわかります。参照カードは、ボタン上にマウスを移動させてしばらくすると表示されます。



このユーザーズガイドで不十分な点は、*Modeling Guide*【モデリングガイド】や *Reference Manual*【リファレンスマニュアル】で補うことができます。モデリングガイドでは、いくつかのダイナミックなモデルの開発方法を記述しており、より応用的なモデリング技法が学べます。リファレンスマニュアルは、Vensim のすべての特徴と操作に関する詳しい情報を提供しています。*Vensim DSS Reference Supplement* は、Vensim DSS 用の文書です。すべてのマニュアルは、オンラインヘルプにあり、トピックを容易に検索することができます。

すべてのマニュアルは、Vensim のホームページから PDF 形式でダウンロードすることができます。マニュアルを印刷したい場合は、この PDF ファイルを利用するとよい。



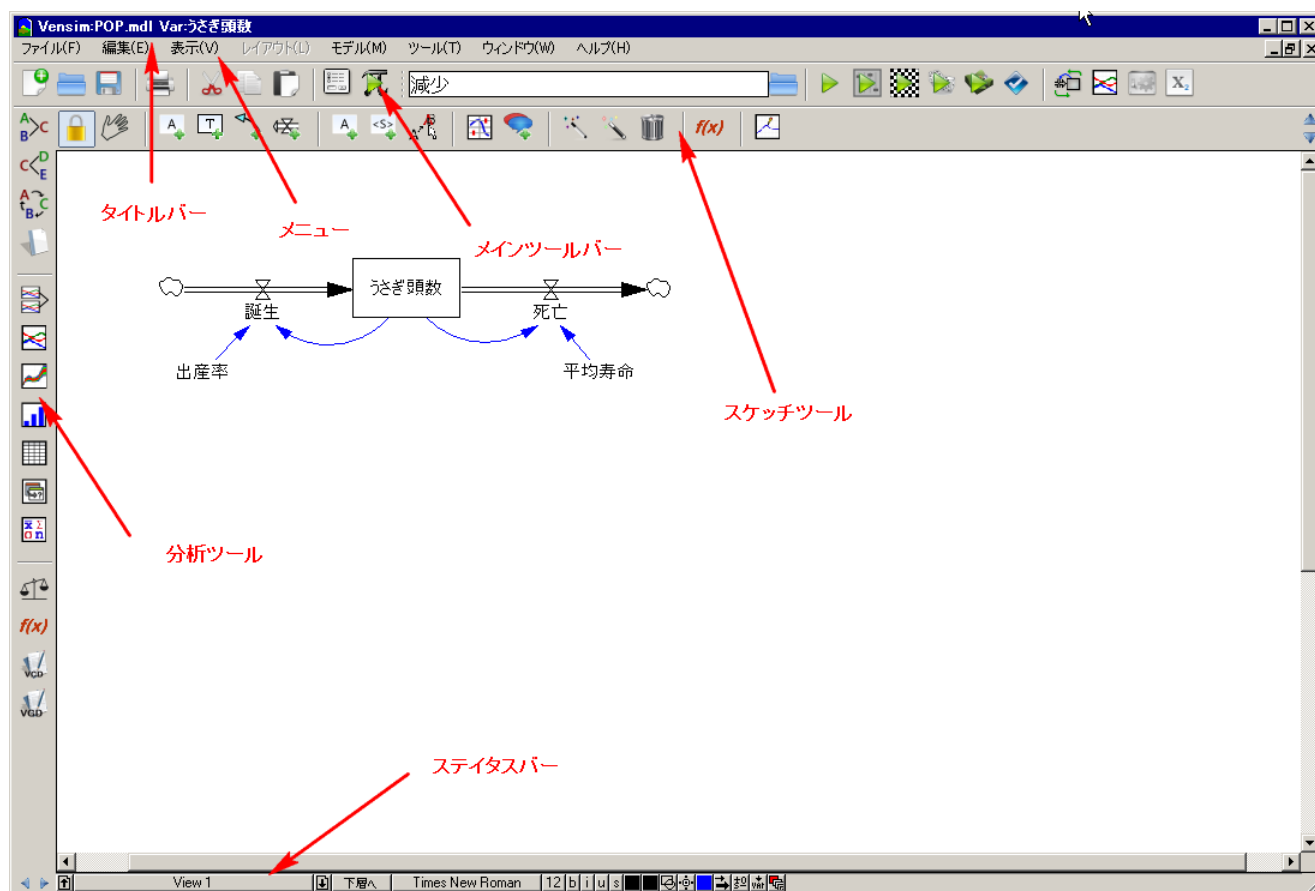


## 第 2 章

# Vensim のユーザインターフェース

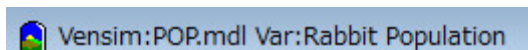
### 2.1 主な特徴

Vensim は、ワークベンチ (プログラム仕事台) とツールセットからなるインターフェースを持ちます。主な Vensim ウィンドウはワークベンチです。ワークベンチはタイトルバー、メニュー、ツールバーおよび分析ツールがあります。下に示されたように、Vensim でモデルを開くと、スケッチツールとステータスバーも表示されます。



## 2.2 タイトルバー

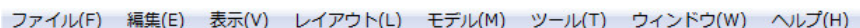
タイトルバーは2つの情報を示します。1つは、開いているモデルの名前 (例えば *POP.mdl*) で、1つはワークベンチで選択された変数 (例えば *Rabbit Population*【ウサギの生息数】) です。



ワークベンチ変数はモデルにおいて選択した任意の変数です。ワークベンチ変数は、その変数の振る舞いを調べたり、より詳しい情報を知るために使われます。ワークベンチ変数は、変数をクリックして選択したり、あるいは制御パネル (本章で後半で紹介する) の変数選択により選択します。

## 2.3 メニュー

Vensim の多くの操作は、メニューから行なうことができます。



- [ ファイル ] メニューには、[ 開く ] [ 保存 ] [ 印刷 ] などの一般的な機能があります。
- [ 編集 ] メニューには、選択部分の [ コピー ] [ 貼り付け ] 機能があります。また変数名の [ 検索 ] ができます。
- [ 表示 ] メニューには、モデルのスケッチを操作するオプションがあります。またモデルのテキスト形式での表示が可能です。テキスト形式での表示は、Vensim Professional と DSS のみ対応しています。
- [ レイアウト ] メニューには、スケッチされた図の位置やサイズを操作する機能があります。
- [ モデル ] メニューには、シミュレーションコントロールやモデル設定を行う機能があります。またモデルのチェックや、データセットの入出力を行います。
- [ ツール ] メニューには、全体のオプションや、分析ツールやスケッチツールを操作する機能があります。Vensim PLE および PLE Plus では、[ ツール ] メニューはなく、[ オプション ] メニューがあります。
- [ ウィンドウ ] メニューには、別のウィンドウへの切り換え機能があります。
- [ ヘルプ ] メニューには、オンラインのヘルプがあります。

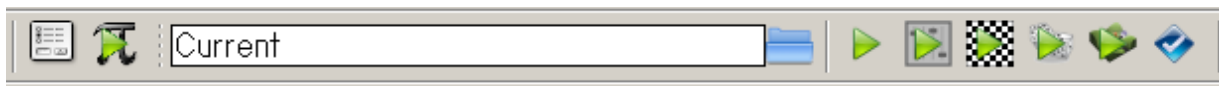
メニューは状況によって変化します。また選択されているウィンドウに対してプログラムが実行されます。よく使われるメニューコマンドはショートカットキーを持っており、さらにツールバーからも実行することが可能です。

## 2.4 ツールバー

ツールバーには、よく使われるメニュー操作やシミュレーション作業のボタンがあります。最初のセットには、[ ファイル ] や [ 編集 ] メニューのボタンがあります。



次のセットには、シミュレーション名の編集ボックスやシミュレーションに用いられるボタンがあります。



最後のものは、Vensim で表示されるウィンドウを操作します。ウィンドウを前面に出したり、操作可能にします。



ツールバーで表示されるボタンは、Vensim のラインアップにより異なります。例えば、PLE では以下のように、簡略化されています。



## 2.5 ウィンドウの種類

Vensim のウィンドウには、いくつかの種類があります。

1. モデルウィンドウは、モデルを作成したり、修正や操作したりするのに用いられます。Professional や DSS では、複数のモデルを同時に開くことができます。
2. 出力ウィンドウは、Vensim の分析ツールによって作成され、グラフや表およびリストからなります。
3. コントロールウィンドウには制御パネルがあります。制御パネルは、Vensim の設定を管理するタブダイアログボックスを持ちます。また Professional や DSS で用いられる下添え字コントロールは、配列を扱います。

### 2.5.1 ウィンドウ間の移動


ウィンドウを選択するか、新たに作成した場合、そのウィンドウはトップへ移動し、操作可能になります。その間、他のすべてのウィンドウは操作できません。ウィンドウを選択し、操作可能にする方法は次のものがあります。

1. ツールバー上の適切なウィンドウボタンをクリックします。
2. ウィンドウ間を循環する **Ctrl**+**Shift**+**Tab** キーを押します。
3. [ウィンドウ] メニューの次の項目を実行します。[モデルウィンドウを前面に]、[モデルウィンドウを背後に]、[制御パネル]、[下添え字] があります。
4. マウスによって適切なウィンドウをクリックします。ただし、これはそのウィンドウが見えている場合にのみ有効です。

最後の方法が、実際よく使われます。

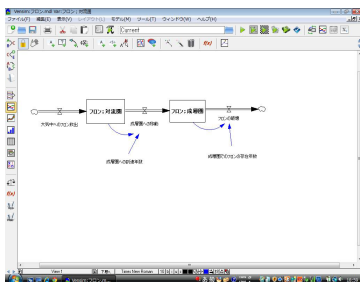
### 2.5.2 同じ種類のウィンドウ間の移動

出力ウィンドウが複数あったり、モデルウィンドウが複数ある場合には、以下の方法でウィンドウを切り換えることができます。

1. 出力ウィンドウボタン  を繰り返しクリックします。
2. **Ctrl**+**Tab** を押します。
3. [ウィンドウ] メニュー→[出力ウィンドウリスト] に示されるウィンドウを選択します。また [ウィンドウ] メニューの下段にあるモデルウィンドウを選択します。
4. マウスによって適切なウィンドウをクリックします。ただし、これはそのウィンドウが見えている場合にのみ有効です。

### 2.5.3 モデルウィンドウ





モデルウィンドウは、モデルを作成するために使用されます。モデルウィンドウには、モデルの構造や方程式を書くスケッチツールがあります。ステータスバーは、スケッチを修飾するためのボタンがあります。また、複数の表示画面を持つことができます。1つの表示画面は、書籍の各ページが物語の一部を構成するのと同じように、モデルの一部を示します。Professional と DSS では、モデルウィンドウをテキスト表示し、モデルを作成したり、編集することができます。その際、ステータスバーはテキスト編集バージョンに変わります。



#### 2.5.4 スケッチツール

















スケッチツールはグループ化されています。Professional や DSS では、スケッチツールを追加したり、ツールの操作を変更したりすることができます。このようにしてカスタマイズされたツールセットは、ファイルに保存され、次回に Vensim を起動した際にも利用できます。組み込まれているデフォルトのスケッチツールセット (*default.sts*) には、モデルを作成するために必要なスケッチツールをほぼ備えています。




PLE には、モデル変数 、変数合体 、上層に戻す 、下層に隠す  がありません。

Professional や DSS では、スケッチツールを右ボタンでクリックすることによって、オプションの変更ができます。ツールの設定を変更した場合、Vensim の終了時に、設定を保存するかどうかの確認が表示されます。新しい設定を保存するには、上書きします。保存できるツールセット数には制限はありませんが、利用しやすいツールセットを 1 つ作るのがよいでしょう。

組み込まれているスケッチツールは次のとおりです。

- スケッチ固定  – スケッチを固定します。マウスで変数を選択し、ワークベンチ変数とすることができます。しかしスケッチの配置は変更できません。
- スケッチ移動  – スケッチされた変数や矢印を移動させたり、サイズを変更します。
- 変数  – 変数（定数や補助変数およびデータなど）を作成します。
- スtock変数  – 箱で囲まれた変数（ストックに用いられる）を作成します。
- 矢印  – 直線、または曲線の矢印を作成します。
- フロー  – フローを作成します。フローや矢印に直交するバルブと変数名からなり、発生と消滅（雲  ）を持つこともあります。
- モデル変数  – すでに存在する変数を追加します。またその変数に影響する変数を追加します。
- 代行変数  – すでに存在する変数を追加します。しかしその変数に影響する変数は追加しません。
- 変数合体 –  2 つの変数を合わせて1 つの変数にします。ストックと雲、矢印の合体にも使われます。
- 入出力オブジェクト  – スライダーやアウトプットグラフおよび表をスケッチに追加します。
- スケッチコメント  – スケッチにコメントや図を追加します。
- 上層に戻す  – 変数をスケッチに表示させます。
- 下層に隠す  – 変数をスケッチで非表示にします。
- 消去  – 構造や変数やコメントをスケッチから消去します。
- 方程式  – 方程式エディタをつかって、モデルの方程式を作成します。

- ・ 参照モード  – 20 章で説明がある参照モードに使用します。

モデルを作るためには、スケッチツールをマウスでクリックします。また、キーボード上の文字を押すことにより、ツールを選択することができます。**1** はスケッチツールの一番左にあるツール、**2** は 2 番目のツール、**0** は 10 番目、**Q** は 11 番目、**W** は 12 番目となります。テンキーではこの操作はできません。また、モデルウィンドウがアクティブな場合のみ、こうしたキーボード操作が可能です。スケッチ画面にマウスを移動させて、一度クリックすれば、選択されたツールが描かれます。ただし、矢印やフロー変数は、一度クリックしたあと、マウスを移動させ、もう一度クリックします。

**注意** スケッチツールの選択は持続性があります。つまり別のスケッチツールを選ぶまで、選択されたスケッチツールは適用可能な状況にあります。

### 2.5.5 ステータスバー

ステータスバーは、スケッチとオブジェクトの状態を示します。ステータスバーには、選択されたオブジェクトの状態を変更し、別の表示形式に変更するボタンがあります。



スケッチの属性のうち、以下のものはステータスバーにより変更することができます。

- ・ 選択された文字列の変更；フォントや文字サイズ、太字、斜体、下線文字、取り消し線。
- ・ 表示レベル（表示、非表示）。
- ・ 変数の色、箱の色、取り囲む図形、文字列の配置、矢印の色、矢印の幅など。

Professional および DSS を用いて、モデルウィンドウをテキスト表示する場合は、ステータスバーがテキスト編集用に変更されます。

### 2.5.6 シミュレーション

モデルウィンドウは、シミュレーション分析にも使用されます。最も重要なことは、モデルウィンドウにおいてシミュレーションの準備や統合シミュレーションが実行できることです。シミュレーションの準備において、すべての定数や表関数は強調表示されます。それらをクリックすることによって、シミュレーションに使用される値を一時的に変更することができます。統合シミュレーションでは、すべての定数にスライダーが表示されます。こうしたシミュレーション分析の詳細は 13 章にあります。シミュレーションの準備や統合シミュレーションを実行するには、ツールバーにあるアイコンをクリックします。

## 2.6 出力ウィンドウ

出力ウィンドウは分析ツールをクリックすることにより作成されます。分析ツールはモデルから情報を集めて、情報を出力ウィンドウに、図やグラフおよび文字列として表示します。アウトプットウィンドウは、いくつも作成されている場合があります。これを一つ



ずつ閉じるには、アウトプットウィンドウの左右にある消去ボタンをクリックします。一度にすべてのアウトプットウィンドウを閉じる場合には、[ウィンドウ]メニュー [全出力ウィンドウを閉じる]を選択します。

### 2.6.1 分析ツール





分析ツールは、ワークベンチ変数の情報を示すために用いられます。分析ツールはツールセットとしてグループ化されています。PLE と PLE Plus では、備え付けのツールセットだけが利用できます。それ以外のラインアップでは、ツールセットは編集可能です。またデフォルトのツールセット (*default1.vts*) と完全なツールセット *default2.vts* があります。

PLE と PLE Plus 以外では、同じ分析ツールに異なる情報を表示させることができます。その設定には、ツールを右ボタンでクリックし、オプションを変更します。またツールセットにツールを追加することもできます。ツールセットに変更を加えた場合、Vensim の終了時に、それを保存するかどうか選択します。またツールセットを読み込むには、[ツール]メニュー [分析ツール] [開く]とします。








以下のツールの説明は、デフォルトよりもツールが多いツールセット *default2.vts* を用います。これに対して、PLE や PLE Plus は利用できるツールが少なくなっています。


参考までに、**[Ctrl]+[1]** は一番上にあるツールを操作可能にし、**[Ctrl]+[2]** は 2 番目を操作可能にします。この操作は最初の 10 個まで有効で、10 番目は **[Ctrl]+[0]** となります。

#### 構造分析ツール





- 因果ツリー  -ワークベンチ変数に影響する変数を樹形図として描きます。
- 波及ツリー  -ワークベンチ変数がどの変数に影響するのかを樹形図として描きます。
- 因果ループ  -ワークベンチ変数をもつフィードバックループを表示します。
- ドキュメント  -ワークベンチ変数の方程式、定義、単位を表示します。

#### データセット分析ツール

- 直接原因グラフ  -ワークベンチ変数に影響する変数の振る舞いをグラフとして示します。
- グラフ  -ワークベンチ変数を 1 つのグラフとして示します。直接原因グラフとは異なるオプションがあります。
- 感度分析グラフ  -1 つの変数の感度グラフを作成します。また感度テストによる不確実性の範囲を作成します。
- 棒グラフ  -ある時点での棒グラフを作成します。または全期間のヒストグラムを作成します。
- 表  -ワークベンチ変数の表を作成します。
- 縦表  -時間軸が縦方向の表を作成します。
- シミュレーションの比較  -最初に読み込まれたデータセットで用いられた定数と 2 番目に読み込まれたデータセットを比較します。

- ・ 統計  -ワークベンチ変数の統計値を示します。



### その他のツール

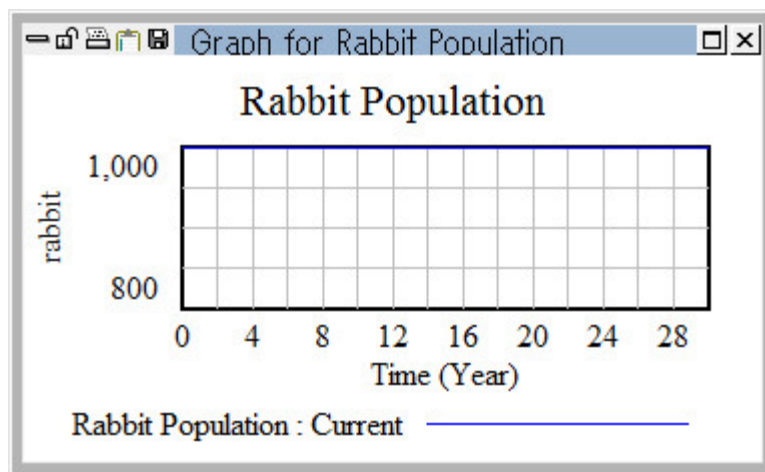
- ・ 単位チェック  - [モデル] メニュー→ [モデルチェック] を行うための手段の1つ。
- ・ 方程式  -方程式を利用するための手段の1つ。
- ・ Venapps 編集  -Venapp の編集をサポートする。
- ・ テキスト編集  -テキストエディタを起動する。それは、*vgd* ファイルを編集します。

ツリー図や直接原因グラフ、感度分析グラフ、表および統計といったツールは、ワークベンチ変数の原因を示すだけでなく、その影響を示すようにすることもできます。

## 2.6.2 分析ツールの出力

分析ツールをクリックすると新しいウィンドウが表示されます。ただし、表とドキュメントは、それまでの出力に追加されていきます。出力されたウィンドウは、削除しない限り残ります。またモデルが変更されたとしても、その結果はアウトプットには反映されません。しかし、12章で紹介する「入出力オブジェクト」を使用して、スケッチへ埋め込まれた出力については、モデルの変更が出力に反映されます。

分析ツールによる出力の一例は、次のようなグラフです。ボタンの仕様は、すべてのアウトプットで共通です。このウィンドウの左上にはボタン  があり、左から [終了] [固定] [印刷] [コピー] [保存] となっています。右上のボタン  では、左側が [最大化]、右側が [削除] です。



- ・ モデルを変更したため、古い出力を削除するには削除ボタン（上段の左右にある）をクリックします。または、**[Del]**キーを押します。
- ・ すべてのアウトプットウィンドウを削除するには、[ウィンドウ] メニュー→ [出力ウィンドウを全て閉じる]を選択します。
- ・ [固定] ボタンにより、そのアウトプットウィンドウを削除されないようにする




ことができます。[ 固定 ] を改めてクリックすることで、ロックは解除されます。

- ・ [保存] や [ コピー ] ボタンによって、アウトプットウィンドウを保存することができます。出力は、クリップボードを通じて、別のアプリケーションに貼り付けする機能です。
- ・ 例え、アウトプットウィンドウを削除してしまった場合でも、ツールボタンを改めてクリックすることで、アウトプットウィンドウを再生成することができます。

分析ツールによる出力は簡単に作成できるだけでなく、削除も容易です。分析ツールは新たに情報を作成しませんが、より有益で、わかりやすい形式で既存情報を表示します。

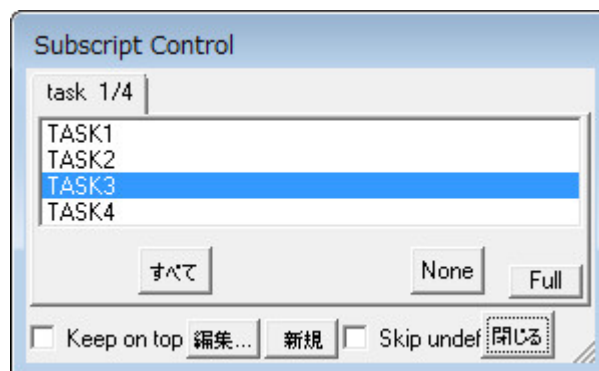
## 2.7 制御パネル

制御パネルは、Vensim 内部のセッティングを変更します。例えば、どのワークベンチ変数が選択されているか、あるいは、どのデータセットが読み込まれているかといった情報を管理します。制御パネルを開くには、ツールバーの制御パネル  をクリックするか、あるいは [ウィンドウ] メニュー→ [制御パネル] をクリックします。制御パネルは、6つのタブを持つフォルダ (Vensim PLE および PLE Plus は5つ) に整理されています。ウィンドウの一番上に配置されているタブをクリックすることにより、フォルダが変わります。




- ・ [変数] 変数を選び、ワークベンチ変数とすることができます。
- ・ [時間軸] 分析ツールが対象とする期間を限定することができます。
- ・ [目盛り] 出力グラフの目盛りを変更することができます。
- ・ [データセット] 保存されているデータセットを管理することができます。
- ・ [グラフ] カスタムグラフを作成します。
- ・ [位置確保] プレースホルダの値をセットするコントロールです。

## 2.8 下添え字コントロール



下添え字は配列を意味します。この例では、「仕事」が「準備」、「設計」、「製造」、「出荷」からなるような場合です。

下添え字コントロールは Vensim Professional と DSS のみ対応しています。それは分析ツールが対象とする下添字の範囲を作成したり、編集したり、選択したりするために用いられます。例えば task【仕事】という変数が下添え字を持ち、今は「TASK3」が選択されている場合、分析ツールは、TASK3 だけの情報を表示します。下添え字の選択は、個別にクリックするか、それとも [すべて] もしくは [None]【すべて選択しない】を利用します。

下添え字コントロールを表示するには、ツールバーの下添え字  をクリックします。新しい下添え字を作成するには、[新規] ボタンを、現在ある下添え字を編集するには、[編集] ボタンをクリックします。

## 第 3 章

# 実際の操作

### 3.1 Vensim によるモデリング

Vensim によりモデルを作る場合、以下のような方法がとられます。

- モデルを構築するか、既存のモデルを開きます。
- 構造分析ツール（樹形図）を使用して、構造を検討します。
- モデルのパラメータを操作して、それがどのように影響するのかをシミュレーションで確認します。
- データセット分析ツール（グラフと表）を使用して、より詳しいデータの振る舞いを検討します。
- シミュレーション実験をおこなって、モデルを改良します。
- モデルとその振る舞いをプレゼンテーションします。その際、統合シミュレーションやカスタマイズされたグラフや表が有効です。

モデルを構築し、それを検討し、修正するという繰り返しの作業が実際には必要です。はじめは、単純なモデルから作成し、実際に稼働するシミュレーションモデルを作成するようにします。実際に稼働するモデルに対して、必要に応じて修正し、改良します。

Vensim にはシミュレーション結果を表示する独自機能があります。シミュレーションの結果は、すべての変数でダイナミックな値として保存されます。また関心のある変数については、分析ツールを用いてその詳細を見ることができます。


#### 3.1.1 労働力と在庫の例

この章では、労働力/在庫モデルを用いて Vensim の操作を行います。このモデルは単純ですが、習得に値するモデルです。このモデルは、在庫管理の方針と雇用方法の相互作用により、生産の不安定が発生することを示します。また採用と解雇に対して積極的であることが、より労働力の安定をもたらすという直感に反した結果も示します。この労働力と在庫モデルは、モデリングガイドの 2 章で作成されています。

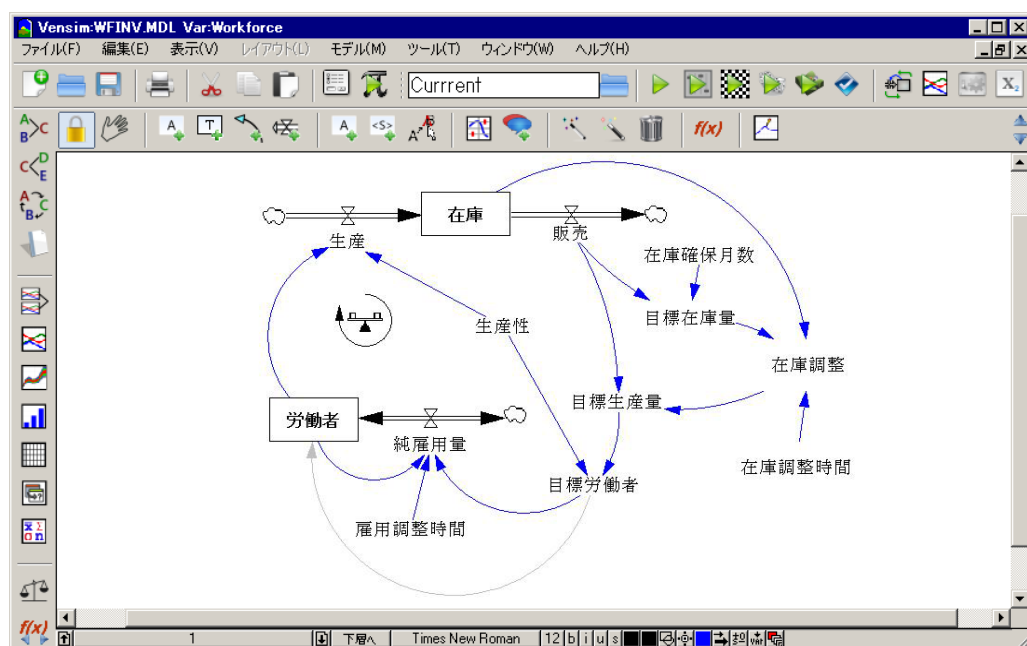
### 3.2 Vensim の開始

- Windows の [ スタート ] メニューの [ スタートプログラム ] [ Vensim ] から Vensim のプログラムを起動します。


### 3.2.1 モデルを開く

- [ファイル] メニュー→ [開く] を選択します。または、ツールバーの開く  を使います。
- *UserGuide\chap03* ディレクトリにある *wfinv.mdl* を開きます。このディレクトリは、比較的新しい OS では、*C:\Users\Public\Vensim\models* であり、場合によっては *C:\Program Files\Vensim\models* になることもあります。

次のような労働力/在庫モデルが開きます。



このモデルは、在庫を持つ製造工場のダイナミックな振る舞いを示します。タイトルバーは、開いたモデル (*wfinv.mdl*) の名前と、ワークベンチ変数 (Workforce【労働力】) を表示します。労働者 という変数は、スケッチにもあります。ワークベンチ変数は、注目したいと考えるモデル中の任意の変数です。別の変数をクリックすることにより、いつでも注目する変数を変更することができます。

- スケッチ固定  はデフォルトで選択されています。スケッチの *Inventory*【在庫】にマウスを動かし、クリックしてください。


タイトルバーにあるワークベンチ変数が 労働者 から 在庫 に変わります。

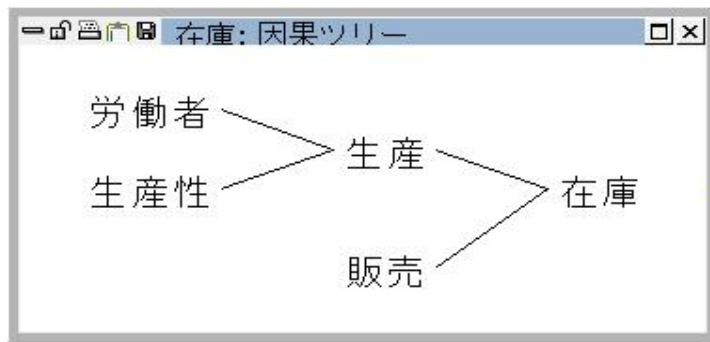
## 3.3 構造の確認

労働力/在庫モデルは比較的単純です。ただし、ストックおよびフロー図の経験がないユーザーには、難しく見えるかもしれません。この視覚的な表現では、矢印は原因と結果を意味します。矢印の後ろに位置する変数は矢印の先にある変数に影響を及ぼします。例えば、*production*【生産】は、*Workforce*【労働者】だけでなく、*productivity*【生産性】




から影響を受けます。

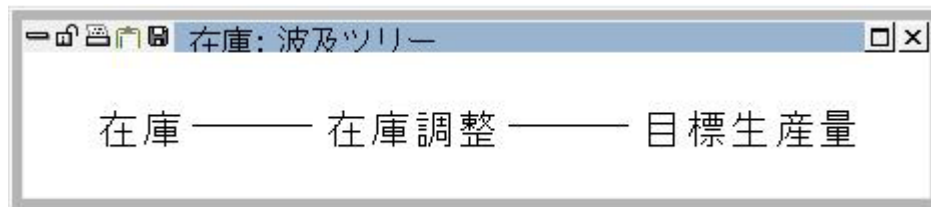
こうしたモデルの構造は、構造分析ツールにより確認することができます。このツールは、モデルのダイナミックな値の振る舞いではなく、構造だけを把握します。より詳しい振る舞いについては、あとで紹介します。

- 分析ツールの一番上にある因果ツリー  をクリックします。出力ウィンドウが表示されます。




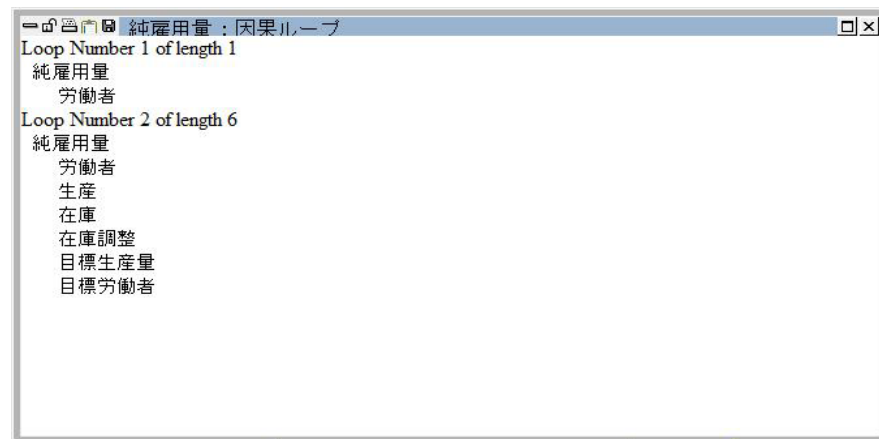
ワークベンチ変数の **在庫** が右側にあり、それに変化をもたらす変数（2段階先まで）が左側にあります。

- この因果ツリーを削除するには、ウィンドウの左上  の左端にある「終了」ボタン、または、右上の  の右側にある「終了」ボタンをクリックします。または **[Del]** をクリックします。
- 波及ツリー  をクリックします。出力ウィンドウが表示されます。



ワークベンチ変数が左側にあり、それが影響を及ぼす変数が右側に（2段階先まで）あります。これらの樹形図は、モデルの情報を2つの方法で示していることになります。スケッチを詳しくみれば、因果関係の全体を見ることができます。樹形図は、モデルの一部を取り出して表示していることになります。

- ツリー図を「終了」ボタン、または **[Del]** キーで削除します。
- マウスでスケッチ上の **net hire rate** 【純雇用量】をクリックし、ワークベンチ変数とします。
- 分析ツールの因果ループ 



出力ウィンドウが表示されます。そこには、ワークベンチ変数 (*net hire rate*) を途中に含むフィードバックループが2つあります。

- ドキュメント ツールをクリックします。




ワークベンチ変数 (*net hire rate*) の方程式とその単位を出力ウィンドウに表示します。

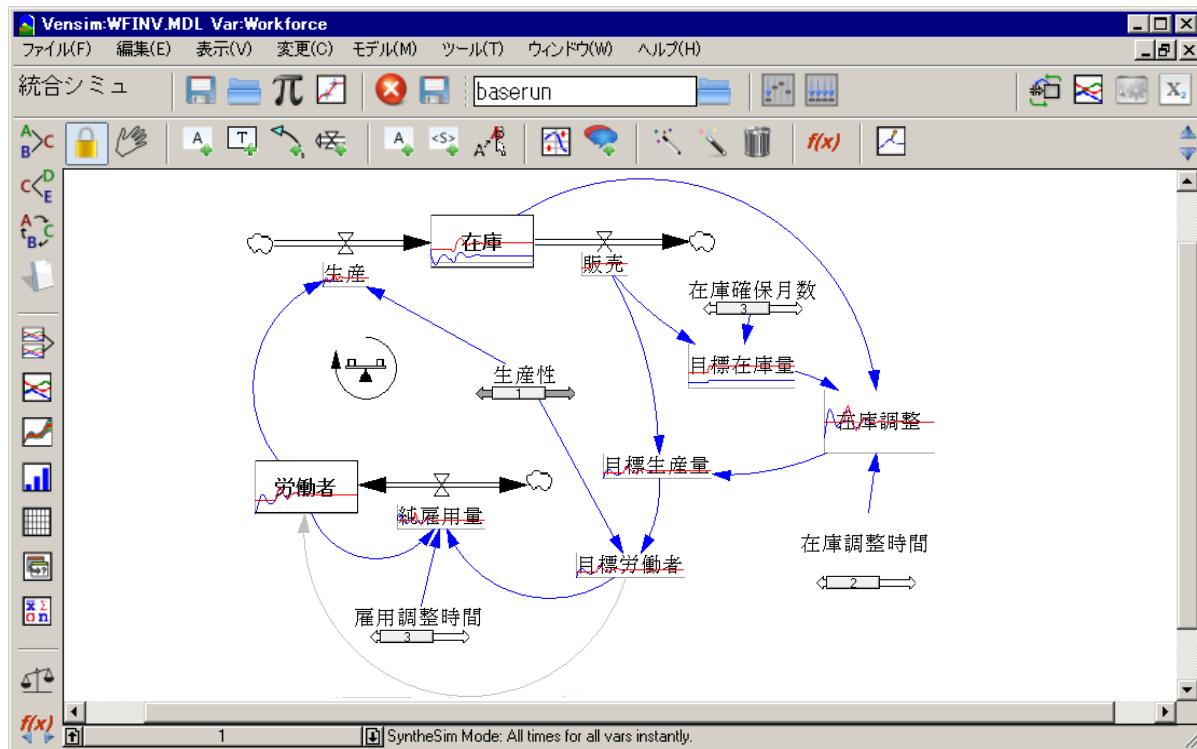
- [ウィンドウ] メニュー→ [全出力ウィンドウを閉じる] をクリックします。

これによって、これまで作成された出力ウィンドウがすべて閉じられます。

### 3.4 モデルのシミュレーション

次に、モデルのダイナミックな値の振る舞いを示します。まず、シミュレーション期間を通じた *Inventory* の推移に注目します。そのためには最初に、モデルをシミュレーションする必要があります。モデルをシミュレーションする最も容易な方法は、ツールバーを使うことです。シミュレーションのセットアップに関する高度なオプションについては、リファレンスマニュアルの8章で述べられているシミュレーションの制御を使います。

- ツールバーのシミュレーション名の編集ボックス  をダブルクリックし、初期設定の *Current* 【現在】 という名前を強調表示し、*baserun* 【基礎実行】 という新しい名前を入力します。この名前は、シミュレーションを実行させた際に、変数の値を保存するデータセットの名前です。
- ツールバーの [統合シミュレーション]  をクリックします。以下のような統合シミュレーションモードに移行します。



変数名の上に重ねられたグラフ、または変数名の下にスライダーが置かれます。スライダーは定数（常に同じ値を持つ変数）に付けられています。また、縮小表示された時間グラフがそれ以外の変数に付けられます。変数名のうえにマウスを移動して、しばらくすると、より大きなポップアップグラフが表示されます。

- ツールバーのシミュレーション名の編集ボックス  をダブルクリックし、その名前を **基礎実行** から **experiment【実験】** へと変更します。こうすることで、シミュレーション結果はデータセット **【実験】** に記録され、基礎実行は変化しません。
- **productivity【生産性】** の下にあるスライダーを前後させます。スライダーを動かすと同時にシミュレーションが実行され、それは青く表示されます。一方で前に保存した **基礎実行** は赤で表示されています。シミュレーションはリアルタイムで実行され、同時にグラフが更新されます。


### 3.5 振る舞いの検討

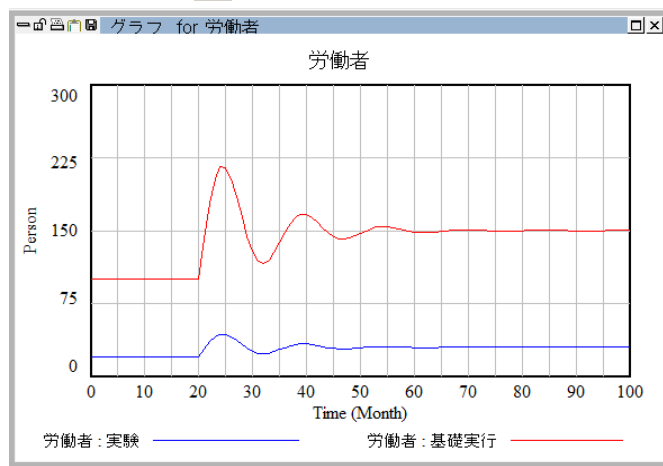
グラフを、より大きく、詳細に表示することもできます。

- マウスを **Workforce【労働者】** に動かして、しばらくそのまましていると、グラフが飛び出してきます。



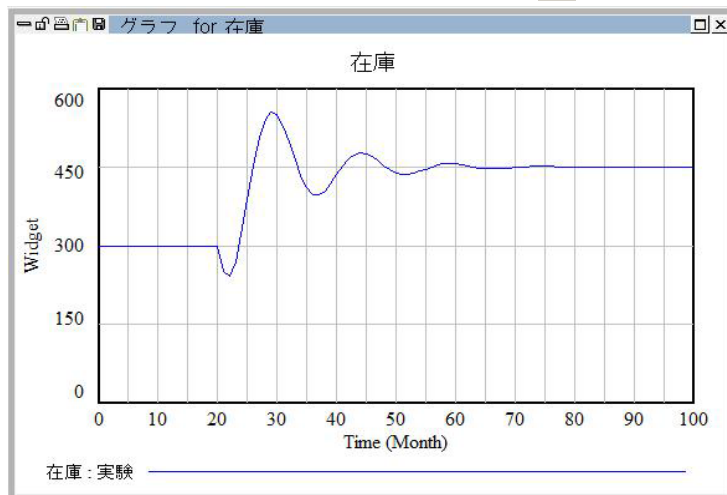
このグラフは大きいだけで、変数名の上に表示される縮小表示グラフと違いはありません。しかし、軸にラベルがあります。

- 労働者をクリックし、ワークベンチ変数にします。
- 分析ツールのグラフ  をクリックします。



こうして作成された2つのグラフは、両者ともに、減衰振動といわれる同じパターンを示しています。また、目盛り以外は同じです。

- グラフを[終了] ボタン、または **[Del]** により削除します。
- 変数 **Inventory** 【在庫】 をクリックし、グラフ  をクリックします。



在庫のグラフは、労働者と同じように振動しています。しかし、在庫は増加の前に、まず減少から始まります。在庫の実際の値を表で確認します。

- 表  ツールをクリックします。



Time (Month)	20	21	22	23	24
"在庫" Runs:	実験	基礎実行			
在庫	300	250	241.667	269.444	322.6
: 基礎実行	300	250	241.667	269.444	322.6

- 出力ウィンドウの下にあるスクロールバー、または や により **在庫** の値を見ます。

**在庫** のグラフは1つだけ描かれます。これは2つのシミュレーション結果が同一であったためです。**生産性** の変化は、**労働者**、*target workforce* 「目標労働者」そして *net hire rate* 【純雇用量】に影響します。このことは、スライダーを動かして図を見れば明確です。なぜなら **生産性** は製品を一単位生産するために必要な労働力であるからです。

- 最新の変更のリセット をクリックするか、**生産性** を元の値に戻すために **Home** を押してください。
- **生産性** 以外の3つの定数についても、スライダーを操作してください。振る舞いを観察して、興味ある結果には詳細グラフを表示してください。

振る舞いを評価する場合、振動の期間、変数が変わる範囲および減衰の程度に注目します。振動の期間は、時間グラフにおける1つのピークから次までの時間です。減衰の程度は、ピークからピークまでの振幅の減少を意味し、この振幅は y 軸の距離になります。

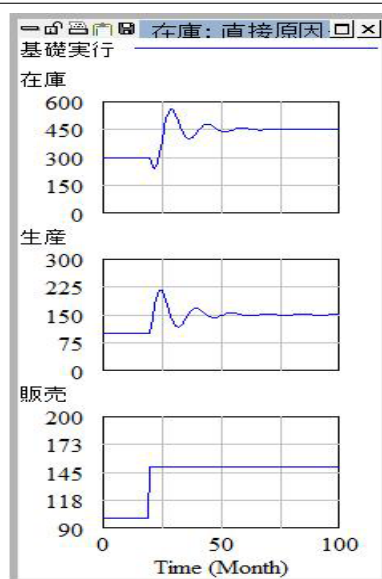
## 3.6 原因追跡

因果ツリーツールをつかって **在庫** を分析した樹形図と同様に、**在庫** に変化をもたらす変数の値をグラフとして見ることができます。

- 全変数のリセット をクリックするか、**Ctrl**+**Home** を押してください。

すべての定数を、元の値に戻します。どの変数でも1本のグラフだけが表示されます。





- **在庫** をクリックして、ワークベンチ変数にします。
- 直接原因グラフ をクリックします。






実行結果とは、*baserun* と *experiment* というシミュレーション結果のことです。

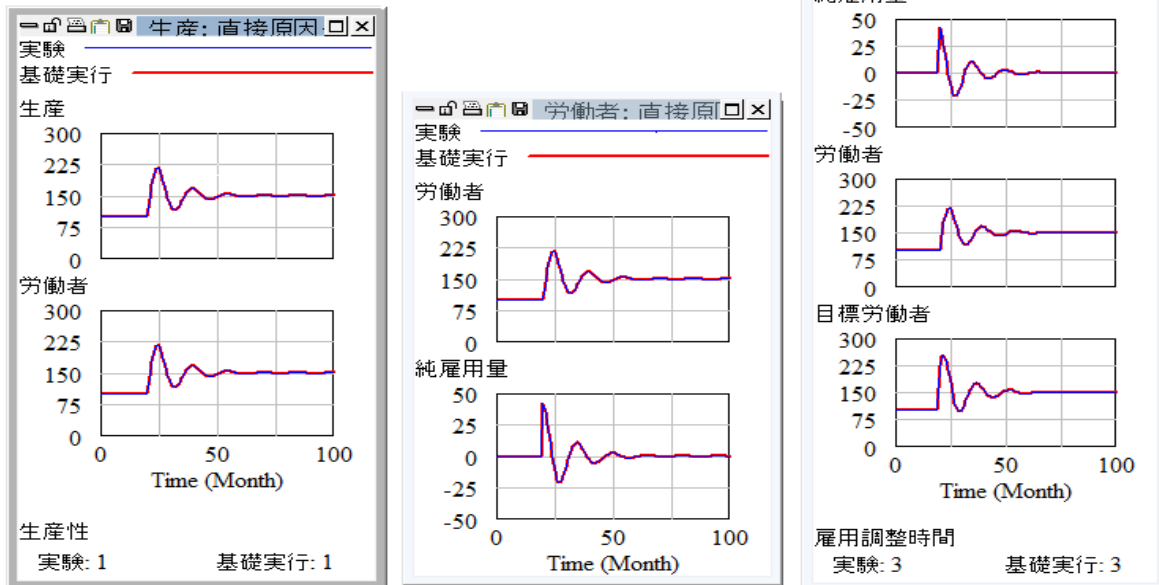
在庫 が一番上にある一連のグラフが表示されます。2段目と3段目は、在庫 に変化をもたらす 生産 性と 販売 です。2つの実行結果は同じであるため、グラフが重なります。[直接原因グラフ] では、グラフを色分けすることや線の太さで区別することができます。

このグラフでは興味深い結果が示されています。在庫 は、一度減少してから、振動し、安定化します。在庫 は、生産 性と 販売 から影響を受けていますが、生産 性 だけが振動しています。販売 は振動していません。したがって、在庫 が振動する理由は、生産 性 にあると考えられます。




モデルのどの部分が、何を引き起こしているのかを確定するうえで、原因追跡は強力なツールです。原因追跡では、因果ツリー  や波及ツリー  および表  も利用されますが、最も一般的なツールは直接原因グラフ  です。それによって、このモデルが振動する原因を探ることができます。

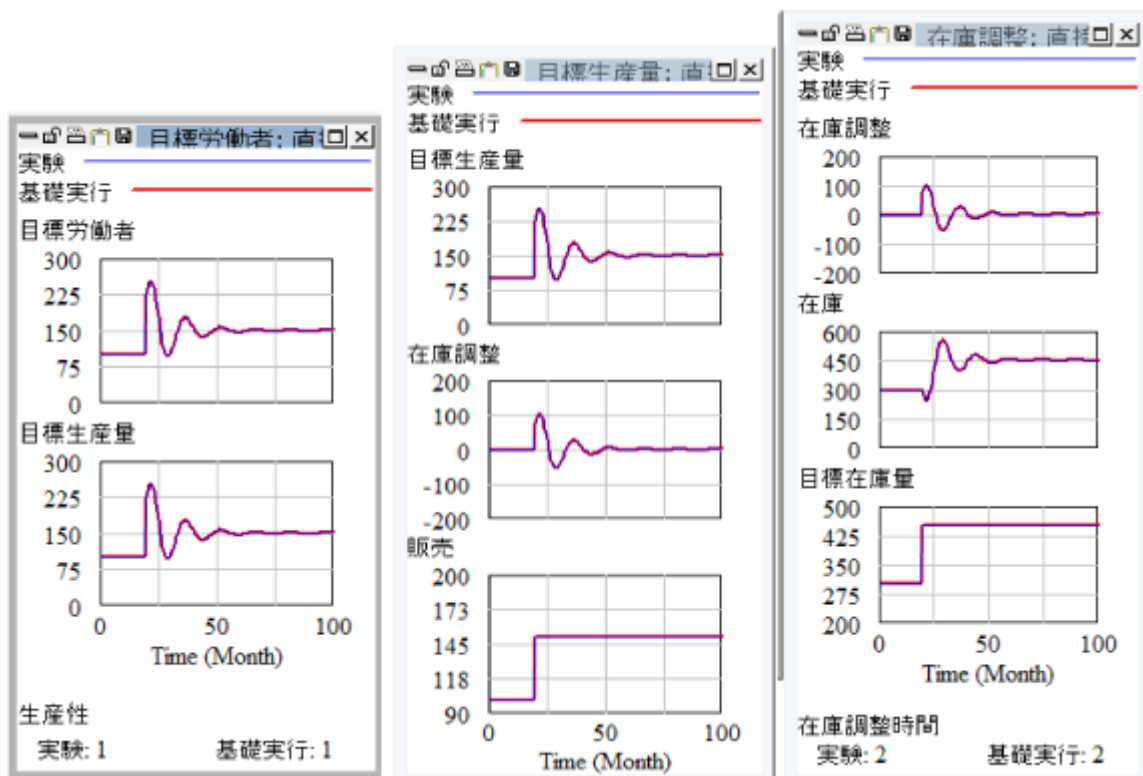
それでは、どのフィードバックループが振動を引き起こしているのかを探します。

- 直接原因グラフにある 生産 性 (文字列の部分) をクリックし、ワークベンチ変数とし、直接原因グラフ  をクリックします。
- 表示された直接原因グラフにある 労働 者 をクリックし、直接原因グラフ  をクリックします。
- 純雇用量 をクリックし、直接原因グラフ  をクリックします。




3つの直接原因グラフが表示されます。振動がこれらの変数によってどのように伝わっているのかを見てください。

- 目標労働者 をクリックし、直接原因グラフ  をクリックします。
- 目標生産量 をクリックし、直接原因グラフ  をクリックします。
- 在庫調整 をクリックし、直接原因グラフ  をクリックします。



最後の2枚のグラフはよく似た振る舞いを示しています。の直接原因グラフは、振動が  ではなく、 に由来していることを示します。 のグラフは、 ではなく、 が振動を起こしていることを示しています。

このよう振動の元は、 にあり、販売 には関係がないことがわかりました。ここで起こっていることを確認するためスケッチを改めて見ます。

➤ ツールバーのモデルウィンドウ表示/入れ替え  をクリックします。

モデルウィンドウを前面に配置し、出力ウィンドウを背面にします。

振動を発生させるフィードバックループを確認します。 から 、、、、、、最後に  へと戻ります。

に注目してください。特に、振動が  ではなく、 を通したフィードバックループによって伝わっていることに注意してください。販売 は、STEP 関数を持つ定数です。販売 は、他の変数に影響しますが、変化を受けることはありません。販売 はいかなるフィードバックループの一部でもありません。販売 は、ストック変数  に、突然の変化（売上の段階的な増加）を伝えます。システム構造（負のフィードバックループ）は、 を補正しようとして、振動を発生させます。これは、ロッキングチェアが、一方に傾くと前後に揺れるのと似ています。


➤ [ウィンドウ] メニュー→[全出力ウィンドウを閉じる]を選択します。

### 3.7 個別のシミュレーション実験

これまで、モデルの振る舞いを調べるために統合シミュレーションを利用しました。また、この機能はモデルを詳しく調べるうえで効率的な方法です。一方で、振る舞いを調べる従来型の方法があります。また、これはシミュレーションごとに、セットアップとシミュレーションを行う方法です。このアプローチは、非常に順序だてられた方法です。そのため、結果を容易に反復できるという長所があります。また、非常に大きなモデルを取り扱う場合では、利用できる唯一の方法でもあります。

➤ ツールバーの停止  をクリックします。

グラフとスライダーは消えて、モデルを最初に開いた状態に戻ります。

➤ ツールバーのシミュレーションの準備  をクリックします。


スケッチにあるいくつかの変数名が、青い背景に黄色の文字となります。これらはシミュレーションを通じて一定の値を持つ定数です。これらに別の値を設定し、モデルへの影響を見ることができます。

➤ スケッチ上で黄色/青で表示された  をクリックします。

編集ボックスが開きます。

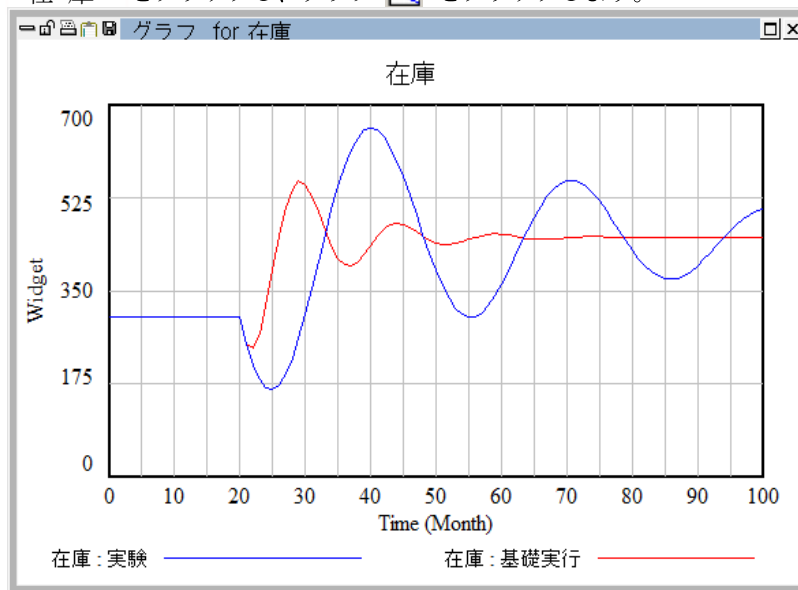
新しい労働者を雇用（または労働者の解雇）する比率を遅らせることで、振動を除去することができるかを実験します。

理想的には、在庫の値をスムーズに増加させたいわけです。

- 編集ボックスにある初期値の「3」を「12」に変更して、を押します。


これにより、雇用調整時間は、3か月から12か月へと変更されます。

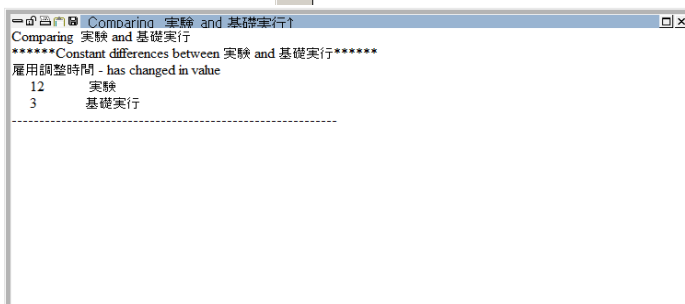
- ツールバーのシミュレーションの実行  をクリックします。シミュレーションが実行され、その値はデータセット 実験 に保存されます。
- 在庫 をクリックし、グラフ  をクリックします。



2つの実験結果が表示されます。1つは基礎実行で雇用調整時間の初期値が3である場合です。2つは実験が持つ雇用調整時間を12にした場合です。

この結果は、雇用や解雇を遅らせることが、振動を大きくし、さらに振動を長引かせることを示しています。


- 分析ツールの比較の実行  をクリックします。



比較の実行ツールは、読み込まれたデータセットの上位2つで用いられる定数や表関数の違いをリストとして表示します。ここでは、基礎実行と実験という2つのデータセットが読み込まれており、その違いは雇用調整時間の値が3と12であることを示しています。

### 3.8 カスタムグラフの作成

1枚のグラフで、重要な変数をすべて見たい場合があります。分析ツールで作られるグラフは、ワークベンチ変数を対象にしました。カスタムグラフを用いることで、変数、データセット、形式を選択した1枚のグラフを表示することができます。カスタムグラフは制御パネルの「グラフ」タブから作成します。

- ツールバーの制御パネル  をクリックし、「データセット」タブをクリックします。



基礎実行は、利用可能なデータセット (Available-Info) のボックスに移動します。

- 読み込まれたデータセット (Loaded-Info) にある 基礎実行 をダブルクリックして、基礎実行を取り除きます。
- 制御パネルの「グラフ」タブをクリックします。

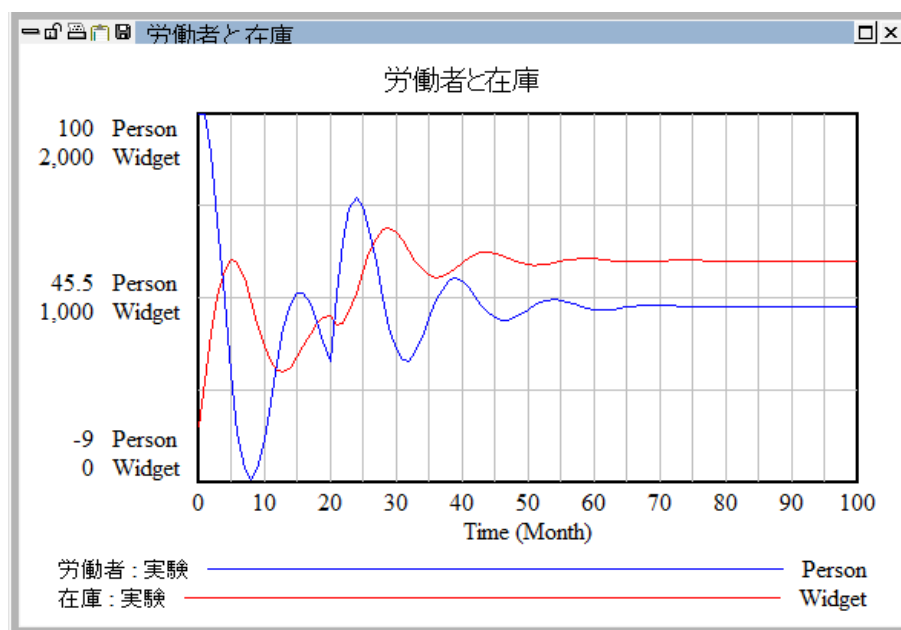


- 「新規」をクリックし、カスタムグラフ編集ダイアログを開きます。

- [タイトル] に、「労働者と在庫」と入力します。
- 画面左下の変数ボックスにマウスを動かし、最上段の「選択」をクリックします。  
変数選択のダイアログが表示されます。


- スクロールバーを下に動かして、「労働者」をダブルクリックします。
- マウスで2段目の「選択」をクリックします。「在庫」をダブルクリックします。  
「在庫」をシングルクリックして、「OK」ボタンを押すことで同様の選択が行えます。

- カスタムグラフダイアログの [OK] をクリックし、画面を閉じます。
- カスタムグラフを表示するには、制御パネルの [グラフ] タブにある [表示] をクリックします。



### 3.9 カスタム表の作成

カスタム表は、変数や表示させる時間を選択できる表です。

- ツールバーの制御パネル  をクリックし、[グラフ] タブをクリックします。
- [新規] をクリックします。
- カスタムグラフ編集ダイアログの [As Table] 【表として作成】をクリックし、カスタム表の編集ダイアログを開きます。



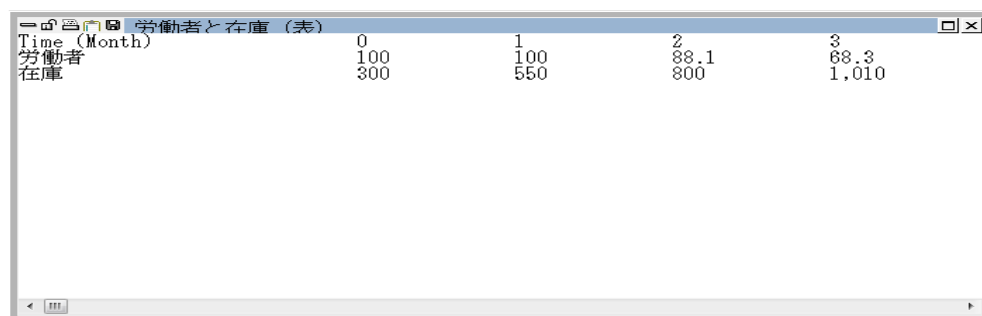
- [タイトル] に「 労働者と在庫(表) 」と入力します。
- [変数] ボタンをクリックし、 労働者 をクリックします。右側にある [追加] ボタンをクリックします。
- 同様に、[変数] ボタンをクリックし、 在庫 をクリックし、[追加] ボタンをクリックします。

次のようなダイアログとなります。

- [OK] をクリックします。

カスタムグラフや表として利用可能なリストが更新されます。作成したカスタム表は、「 労働者と在庫(表) 」と表示されます。 リストに表示する名前は、カスタム表の編集ダイアログの [表の名称] で再入力することもできます。

- リストにある 「 労働者と在庫(表) 」 をダブルクリックします。



Time (Month)	0	1	2	3
労働者	100	100	88.1	68.3
在庫	300	550	800	1,010

### 3.10 要約

単純なモデルを実際に操作しました。これらの操作は、Vensim によるモデル分析の基礎です。たとえ、非常に複雑なモデルであっても、これらの分析ツールは、モデルを理解し、修正するうえで有効です。これから続く 6 つの章は、モデルを作成するための基礎技術を扱います。そのあとで本章で紹介した分析やレポートについてのトピックを改めて紹介します。

## 第 4 章

# 因果ループ図

### 4.1 Vensim モデル





本章は因果ループ図を説明します。因果ループ図は、変数のつながりが因果関係を示すためにそのように呼ばれます。A から B を結ぶ矢印は、A が B に影響することを意味します。因果ループ図は、構造の概念化や理解にとって役立ちます。実際、シミュレーションモデルが作成されていなくても、因果ループ図が有効な場合があります。本章の大部分は操作方法の説明です。たとえ因果ループ図を利用しないのであっても、モデル作成において必要なものです。



因果ループ図はシステムにおける蓄積（ストックやフロー）を扱いません。ストックやフローの作成は 5 章で行います。もしストックやフローに関心がある場合であっても、この章から取り組むことを勧めます。なぜなら、図を描く基本的な方法は同じであり、本章はそれを詳しく記載しているからです。





重要な注意点として、因果ループ図やストックおよびフロー図は、シミュレーションモデルではないことがあります。3 章で使用したモデルがシミュレーションモデルであり、すべての変数間に代数的な関係があります。6 章の「シミュレーションモデルの構築」において、具体的なシミュレーションモデルの作成方法を説明します。Vensim Professional あるいは DSS では、図を用いたモデル作成ではなく、方程式の直接入力も可能です。しかしながら通常は、図を用いてモデルを作成します。


#### 4.1.1 スケッチを描く


一度スケッチツールを選択すると、別のツールを選択するまでは、そのツールが選択状態にあります。マウスのシングルクリック（押して離す）によりスケッチにツールが描かれます。

スケッチ固定  は、マウスカーソルになります。スケッチ固定  は、スケッチされた対象を選択（白黒反転表示される）したり、オプションの変更に用いられます。スケッチされた対象は、スケッチ固定  が選択されていると、動かすことはできません。また、スケッチ固定  は、**[Esc]** キーやキーボードの **[1]** でも選択できます。

スケッチ移動  は、スケッチされた対象（変数や矢印）を移動したり、大きさや形状を変えたりします。また、スケッチ移動  以外のツールも、スケッチされた対象を移動することができます。

変数を描くツール（変数  やストック変数  など）とフロー  を描くツールがあります。これらのツールには、その変数名を入力するための編集ボックスが表示されます。スケッチコメント  は、ダイアログを表示します。

矢印  は、矢印の始点となる単語をクリックし、矢印の終点となる単語をクリックします。始点と終点の間で、マウスクリックを追加すると、曲線となる矢印の中間点を追加します。

**注意** 矢印を引く際に、単語をクリックし、ドラッグ（マウスボタンを押したまま動かす）しないでください。これはクリックした単語を移動させることになります。これはフロー  でも同じです。

### 4.1.2 マウスのヒント

- ・ マウスボタンのクリックとあれば、左ボタンを意味します。

## 4.2 因果ループ図の作成


ここでは、建設計画における因果ループ図を作成します。重要な概念はプロジェクトにおいて残されている仕事量です。図では、この概念が中心になります。最初に、プロジェクトを完了させるために必要な要素をまとめた表示画面を作ります。この表示画面は本の1ページのように、モデルの一部となります。モデルは複数の表示画面を持つことができます。システムについての知識を組み入れるために、別の表示画面を付け加えることもできます。

### 4.2.1 プロジェクト・モデル（project.mdl）

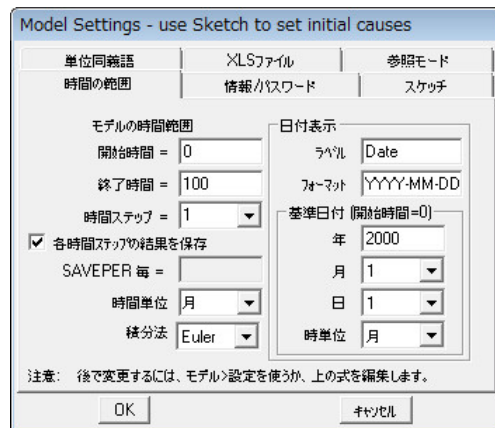
このモデルは、両立しないフィードバックループを持つプロジェクトです。因果ループ図は、「達成すべき仕事」と「超過勤務要求時間」の関係に加えて、超過勤務が「達成した仕事」や「疲労」に与える影響を考えます。当初は、労働力が一定であると仮定します。

- Vensim を起動します。

Vensim は、前回作業した最新のモデルを開きます。


- [ファイル] メニュー→[新規モデル] を選択するか、新規モデル  をクリックします。

次のようなモデル設定のダイアログが表示されます。





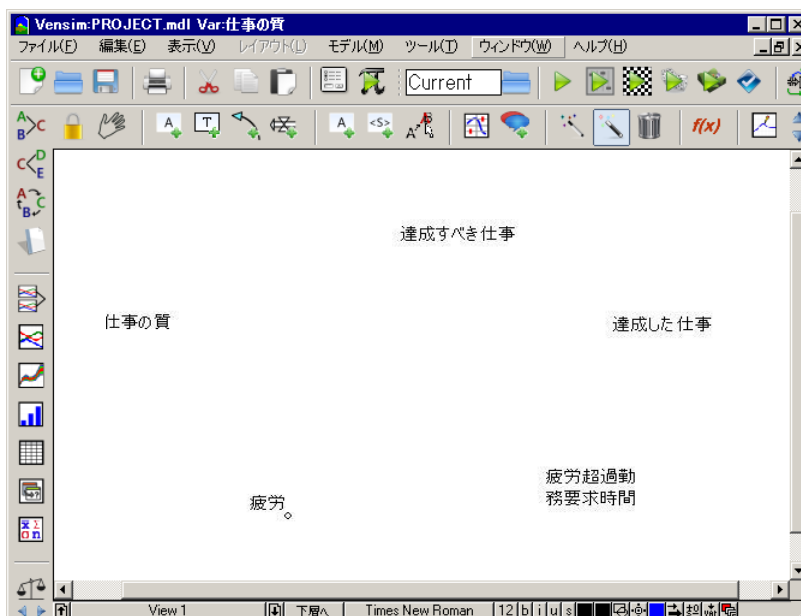
- デフォルトでよいので、[OK] をクリックします。

因果ループ図は「時間の範囲」タブを使用しません。しかし、シミュレーションモデルは「時間の範囲」タブの設定を必要とします。すべての Vensim モデルは、例えそれが使用されなくても、「時間の範囲」タブの設定値を持ちます。


- ツールバーの保存  をクリックし、モデルを保存するディレクトリとファイル名 (project) を指定し、[保存] をクリックします。


### 変数の追加


- 変数  をクリックします。またはキーボードの [3] を押します。
- スケッチの中央上をクリックし、編集ボックスに、「達成すべき仕事」と入力し、 キーを押します。
- スケッチを再びクリックして、以下のような図を描きます。



### スケッチされた対象の移動


- スケッチ移動  をクリックします。またはキーボードの[2]を押します。マウスを移動させたい変数上にもっていき、ドラッグすることで新しい場所に移動します。（キーボードのショートカット操作については半角で行ってください）

ただし、スケッチ移動  以外のスケッチツールであっても、対象を移動することができます。




- 今度は、変数  をクリックします。またはキーボードの[3]を押します。マウスを移動させたい変数上にもっていき、ドラッグすることで新しい場所に移動します。


このようにして、因果関係において重要だと思われる変数を配置しました。

### 矢印の追加




- 矢印  をクリックします。またはキーボードの[5]を押します。「達成すべき仕事」を一度クリックします。ここで、マウスを動かさずに、ボタンをあげてください。マウスを「超過勤務要求時間」まで動かして、クリックします。直線の矢印がこの2つの変数間で描かれます。
- 「超過勤務要求時間」をもう一度クリックし、「成された仕事」をクリックします。直線の矢印がこの2つの変数間で描かれます。

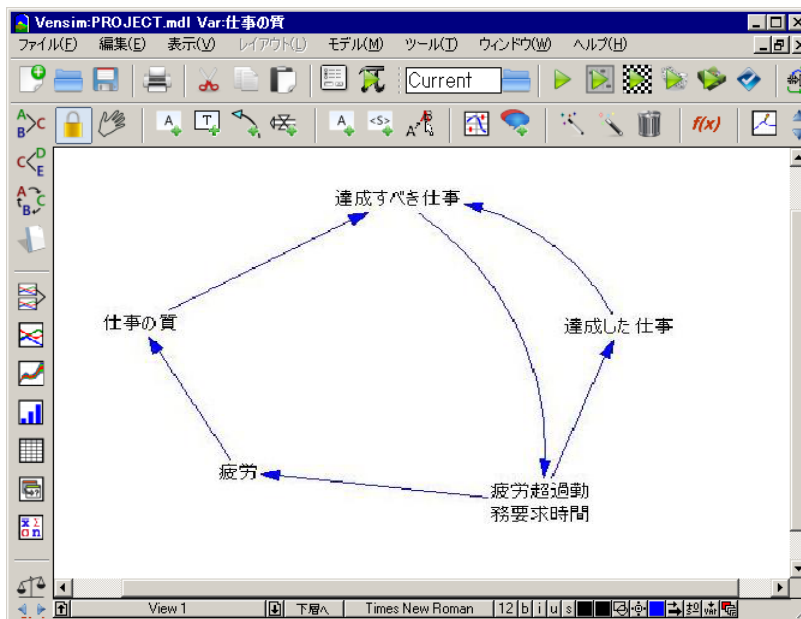
### ハンドル

ハンドルは、矢印の中間にある小さな円です。また箱の角にもあります。さらにフローのなかにもあり、スケッチのいたるところにあります。ハンドルは、サイズの変更や移動に利用します。ハンドルが表示されるのは、変数を作成したとき、矢印を作成したとき、またはスケッチ移動  を選択したときです。ハンドルはスケッチ移動  だけでなく、スケッチ固定  を除くすべてのツールで操作することができます。


- ハンドルを表示するため、スケッチ移動  をクリックします。

### 曲がった矢印


- 「超過勤務要求時間」から「成された仕事」への直線矢印の中間にあるハンドル上にマウスを動かします。マウスを押して、ドラッグすることで、曲線になります。
- 矢印  をクリックします。「成された仕事」を一度クリックし、「成すべき仕事」の右上付近の空白のスケッチを一度クリックします。そして「成すべき仕事」を再度クリックします。曲がった矢印により、変数が結ばれます。ハンドルを矢印  か  でドラッグすることで、曲線の形状が変化します。
- 次の図のように変数を曲がった矢印で結びます。直線で結んだものを曲線にしたり、中間点を置いて作成します。






### 変数の編集

- 変数名を編集するには、変数  を用いてクリックし、編集ボックスに新しい名前を入力します。

### 変数の削除

変数を削除する場合は、[編集] メニュー→[切り取り] を選択します。または **[Del]** を用います。この2つの方法は、削除するかどうかを確認するダイアログが表示されます。削除  を利用することもできます。この場合、確認されることなく削除されます。


[編集] メニュー→[切り取り] や **[Del]** を用いた場合、削除の方法として **[この表示から除去する。モデル構造は変更しない。]** を選ぶことができます。それを選択した場合、スケッチの表示にはありませんが、モデルに存在する変数となります。このオプションには注意してください。

- 変数  を選択し、スケッチに「一時的な変数」を作成し、 を押してください。
- 削除  により、「一時的な変数」をクリックします。

### 取消しとやり直し

モデル作成における失敗は、[編集] メニュー→[元に戻す] や [編集] メニュー→[やり直し] で修復できます。前者は **[Ctrl]+[Z]**、後者は **[Ctrl]+[Y]** でも操作可能です。取り消しとやり直しは、さかのぼることができます。

### モデルの保存

- モデルを保存するには、保存  や [ファイル] メニュー→[保存]、または **[Ctrl]+[S]** を用います。

モデルはデフォルトで、拡張子が`.mdl`というファイルで保存されます。これはテキスト形式です。またバイナリフォーマットの拡張子が`.vmf`で保存することもできます。バイナリフォーマットで保存されたモデルは Vensim Model Reader で利用することができます。バイナリフォーマットのモデルは、テキスト形式モデルより速く開くことができます。しかしこのガイドで紹介する小さいモデルでは、それを実感することはありません。






### 4.3 図形の修正

図形には変更可能なオプションがあり、2つの方法で変更することができます。

1. スケッチされた対象を右クリックする。
2. スケッチされた対象を選択し、ステータスバーでオプションや属性を選択する。

#### スケッチされた対象の選択


1つ、または複数の選択には、いくつかの方法があります。

- ・ スケッチ移動  で対象をクリックします。
- ・ 複数の対象を選択するには、スケッチ移動  またはスケッチ固定  を用いて、スケッチ上で、それらを囲むように左ボタンを押したままマウスをドラッグします。
- ・ 複数の対象を選択するには、スケッチ移動  を用いて、**[Shift]**を押したまま、対象をクリックします。
- ・ スケッチ上のすべての対象を選択するには、[編集] メニュー→[全て選択] を選びます。これは**[Ctrl]+[A]**でも可能です。
- ・ 選択された対象を、選択から外す場合は、スケッチ移動  を選択し、**[Shift]**を押したまま、外す対象をクリックします。
- ・ すべての対象を選択から外すには、スケッチの空白領域をクリックします。

#### 4.3.1 スケッチのレイアウト

スケッチのレイアウトを整える方法があります。例えば、対象をデフォルト値の大きさに揃える、最後に選択された対象の位置やサイズにそろえるなどのものです。

「達成すべき仕事」が中心となるように、変数の配置を整えます。

- スケッチ移動  を選択します。または**[2]**をクリックします。
- 「超過勤務要求時間」をクリックし、**[Shift]**を押しながら「成すべき仕事」をクリックします。[レイアウト] メニュー→[中央に揃える] を選択します。

「超過勤務要求時間」は、「達成すべき仕事」の真下に移動します。

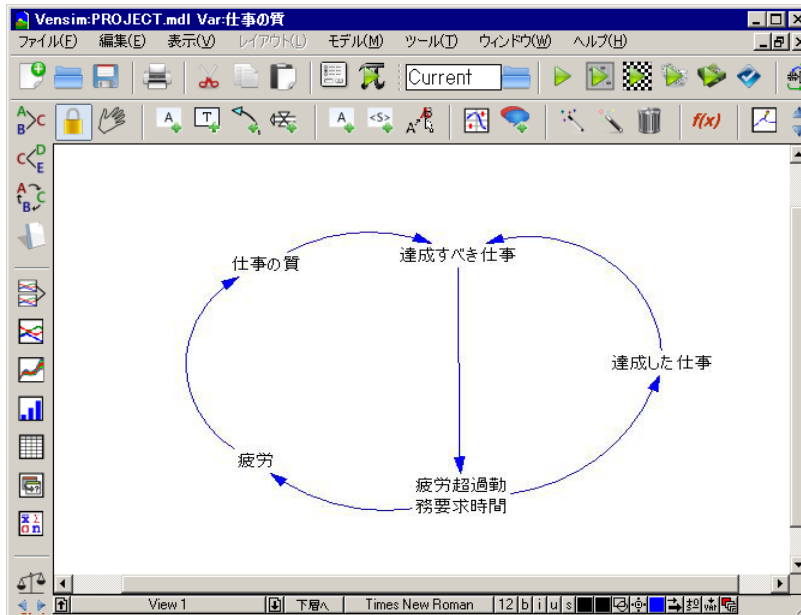
- 「仕事の質」をクリックし、**[Shift]**を押しながら「達成すべき仕事」をクリックします。[レイアウト] メニュー→[垂直位置を揃える] を選択します。
- 「疲労」をクリックし、**[Shift]**を押しながら「仕事の質」をクリックします。[レイアウト] メニュー→[中央に揃える] を選択します。

(キーボードのショートカット操作については半角でお願いします)




- 「疲労」をクリックし、**[Shift]**を押しながら「超過勤務要求時間」をクリックします。[レイアウト]メニュー→[垂直位置を揃える]を選択します。
- 「成された仕事」を、「成すべき仕事」と「超過勤務要求時間」の中間から少し離れた場所に移動させます。
- 矢印を円に近い形へと整形します。

次のようなスケッチが描かれます。

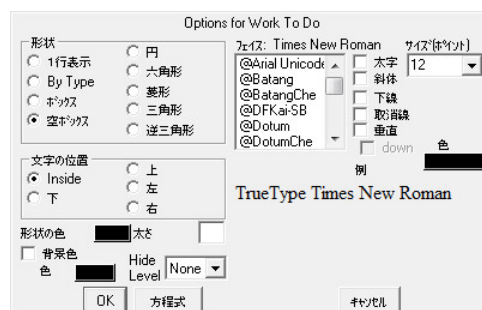


### 4.3.2 スケッチオプション

#### 変数

- スケッチ固定  を選択します。右ボタンで 達成すべき仕事 クリックします。

オプションのダイアログが表示されます。



- フォントやフォントサイズ、色などを変更して、**[OK]** をクリックします。

フォントを赤に変更しました。

[文字の位置] は、[形状] において、[1 行表示] 以外が選択されている場合のみ有効です。

- [編集] メニュー→ [全て選択] を選びます。または **Ctrl**+**A** を押します。ウィンドウの下にあるステータスバーのフォントサイズボタン（おそらくフォントは 12 となっています）をクリックし、14 をクリックします。強調表示されている箱の外側をクリックします。

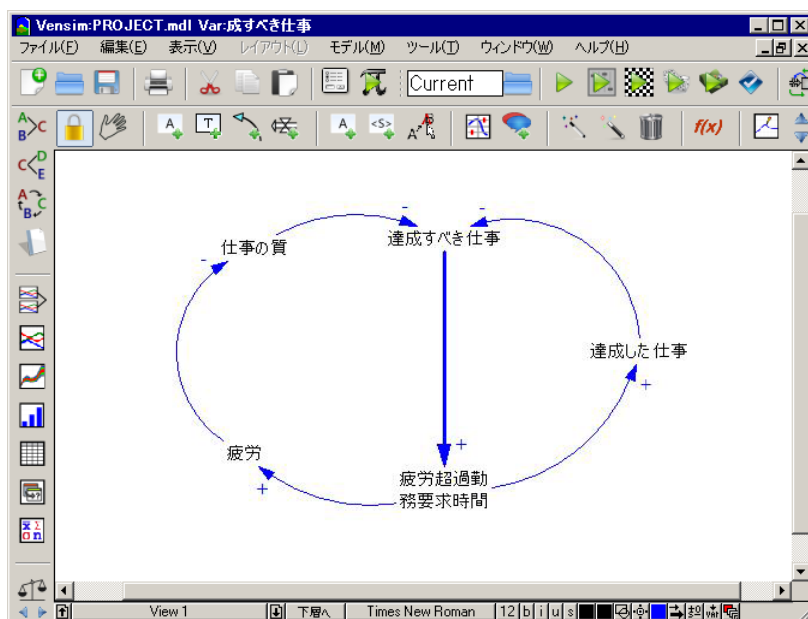
### 矢印

- 「達成すべき仕事」から「超過勤務要求時間」への矢印の先を右クリックします。次のダイアログが開きます。




「達成すべき仕事」が増加すれば、「超過勤務要求時間」も増えるので正の作用があります。



- [Polarity]【極性】の **[+]** をクリックし、[ 極性マークの位置 ]【矢印の外側】をクリックします。[OK] をクリックします。  
極性 (+) は、デフォルトで、矢印の内側に描かれます。
- 以下の図のように極性を変更してください。





次に、正のフィードバックループを太くし、色をつけます。

- スケッチ移動  を選択します。
- 「成すべき仕事」から「超過勤務要求時間」への矢印の先頭を左クリックします。  
 を押しながら、他の矢印の先頭をクリックします。
  - 「超過勤務要求時間」から「疲労」
  - 「疲労」から「仕事の質」
  - 「仕事の質」から「成すべき仕事」


すべてのハンドルが強調表示され、選択された矢印を取り囲む点線で囲まれた箱が表示されます。


- を離します。
- ステータスバーの 2 種類の幅の違う矢印が描かれたボタン  を探します。それをクリックし、上から 5 番目の線を選択します。強調表示された矢印の幅が大きくなります。
- 矢印幅ボタンの左にある色のボタン  （おそらく青色で表示されています）をクリックし、赤に変えます。点線で囲まれた箱の外側をクリックし、矢印の選択を解除します。


### 4.3.3 コメントと画像の追加

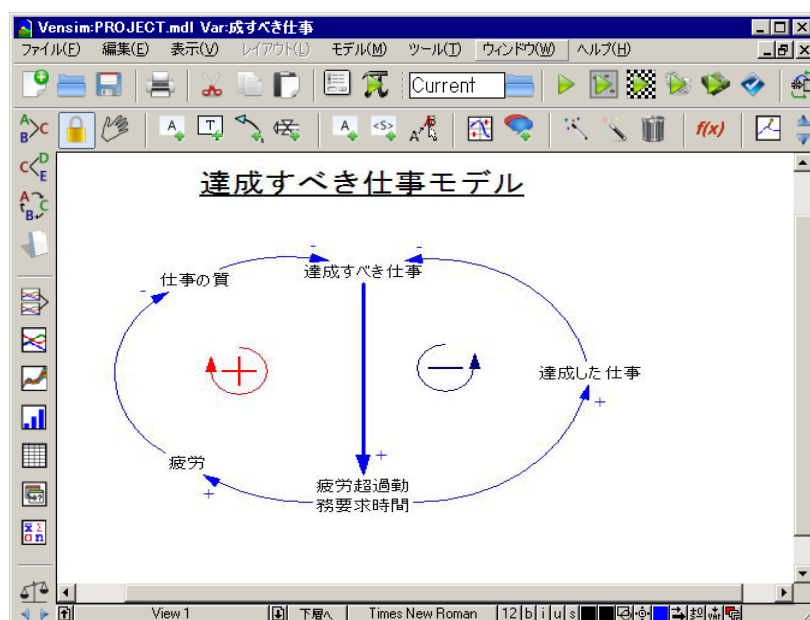
- 図形の上の方にスペースを空ける必要があるなら、スケッチ移動  を選択し、[編集] メニュー → [全て選択] を選びます。これは  +  でも可能です。次に、カーソルを使用して、タイトルを記載する場所を空けるために図形を下げます。
- スケッチコメント  を選択します。タイトルを加えるにはスケッチの一番上をクリックします。次のようなコメントダイアログが開きます。



- スケッチのタイトル「達成すべき仕事モデル」をコメント欄に入力して下さい。またタイトル用のフォント、サイズ、色、形状および文字の位置を選んで、[OK] をクリックします。
- スケッチコメント  を選択したままで、左側のループの中心をクリックします。

[Graphics] の [イメージ] にあるドロップダウンボックス [▼] をクリックし、正を示す (+)、または雪のボール  を選択します。次に、[形状] を [時計回ループ] とします。

- 中央右にある [色] の下にある黒いボタンをクリックし、赤を選択します。同じように [枠の色] も変更し、[OK] をクリックします。必要があればハンドルを動かして、ループ画像の移動や、サイズを変更します。イメージ画像には、クリップボードを使ったビットマップやメタファイルの画像を利用することもできます。
- 右側のループの中心をクリックします。同じように、[Graphics] の [イメージ] にあるドロップダウンボックス [▼] をクリックし、負を示す (-)、またはバランス  を選択します。次に、[形状] を [反時計回ループ] とします。また [色] と [枠の色] を青に変更し、[OK] をクリックします。






#### 4.3.4 モデルの改良




「超過勤務要求時間」が「達成した仕事」を増やし、それが「達成すべき仕事」を減らすという負のフィードバックループがあります。しかし一方で、「超過勤務要求時間」は、「成すべき仕事」を増やす正のフィードバックループも持ちます。

このモデルをより現実的にするために、雇用と解雇によって労働力の調整が可能であると仮定します。雇用は超過勤務の必要性を軽減し、「疲労」の増加を抑えることができます。

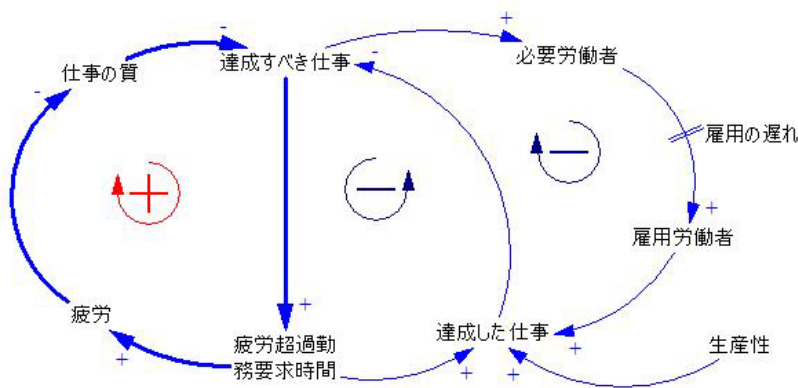
##### フィードバックループの追加

- スケッチ移動  を選択して、「達成すべき仕事」をスケッチの下へ移動します。下図のように、「達成した仕事」にある矢印を整え、負のループ画像を移動させます。
- 変数  を選択します。「必要労働者」、「雇用労働者」そして「生産性」を追加しま

す。このとき変数名を入力後、で追加できます。

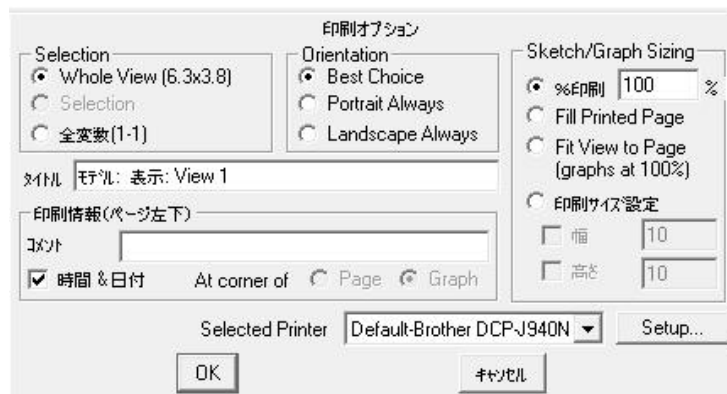
- 矢印  を用いて、変数を矢印で結びます。
- 矢印の極性を追加します。また、「必要量動者」から「雇用労働者」への矢印には、[Delay mark]【遅れの印】にチェックを入れます。
- スケッチコメント  を選択します。遅れの印の上に、「雇用の遅れ」を追加し、青色にします。
- スケッチ移動  を選択し、負のループ画像をクリックし、強調表示します。次に[編集]メニュー→[コピー]（または $\text{Ctrl} + \text{C}$ ）を選択します。さらに[編集]メニュー→[貼り付け]（または $\text{Ctrl} + \text{V}$ ）を選択します。ダイアログで[Replicate]【複製】が選択されている状態で、[OK] または  $\text{Ctrl} + \text{Enter}$  とします。新しいループ画像は、複製元の画像の上に貼り付けられます。コピーされた画像を新しいフィードバックループの中心に移動します。この画像を右クリックし、[形状] を「時計回ループ」に変更し、[OK] をクリックします。

### 達成すべき仕事モデル




## 4.4 スケッチの印刷

スケッチは、印刷ボタンをクリックするか、[ファイル]メニュー→[印刷]を選ぶことで印刷が可能です。



印刷オプションダイアログのオプションの中で重要なものをいくつか挙げます。


- [ Selection ] 画面全体の印刷、指定した部分を印刷、すべてを印刷
- [ Orientation ] 用紙のタテ、ヨコを選択します。
- [ Sketch/Graph Sizing ] 印刷サイズを選択します。1 ページに収まるように縮小印刷することもできます。
- [ タイトル ] 用紙の上部に印刷されるタイトルです。

スケッチは、[ 編集 ] メニュー [ すべて選択 ] や、スケッチ固定  で変数を選択したのち、[ 編集 ] メニュー [ コピー ] で PC のクリップボードに出力することも出来ます。このようにメタファイルとしてコピーされたスケッチの情報は、他のアプリケーションに貼り付けて利用することができます。


印刷については、リファレンスマニュアルの 16 章で、より詳しく説明します。

## 4.5 分析ツール

Vensim の分析ツールは、構造分析ツールとデータセット分析ツールの 2 つがあります。構造分析ツールは、モデルの構造を調べることができます。データセット分析ツールは、変数の振る舞いを決定するデータセットを調べます。この節では、モデルの構造を分析します。利用する構造分析ツールは、樹形図ツール ( 因果ツリーと波及ツリー )、ループツール、ドキュメントツールです。

分析ツールはワークベンチ変数の情報を表示します。ワークベンチ変数を選択するには、変数をクリックします。これが一般的な方法です。通常、変数は、モデルがテキスト表示でなければ、スケッチのどこかにあります。また、樹形図や直接原因グラフにより出力される出力ウィンドウの変数名を使うこともできます。もう 1 つの方法は、制御パネル  をクリックし、[ 変数 ] タブのリストから変数を選択することです。ワークベンチ変数は常にモデルのタイトルバーに表示されます。

**注意** データセットが必要なツールを利用する場合、まだデータセットが作成または読み込まれていないと、[ No runs are loaded. Please load runs. 【データセットが読み込まれていません。読み込んでください】 ] というメッセージが表示されます。これは、シミュレーションを実行する必要があることを示しています。シミュレーションモデルの作成については、5 章で説明します。

- もし、*project.mdl* を作成していないのであれば、*guide/chap04/complete* から、*project.mdl* を開きます。
- スケッチ固定  を選択します。マウスを「成すべき仕事」に移動し、クリックします。タイトルバーに選択したワークベンチ変数の「成すべき仕事」が表示されます。

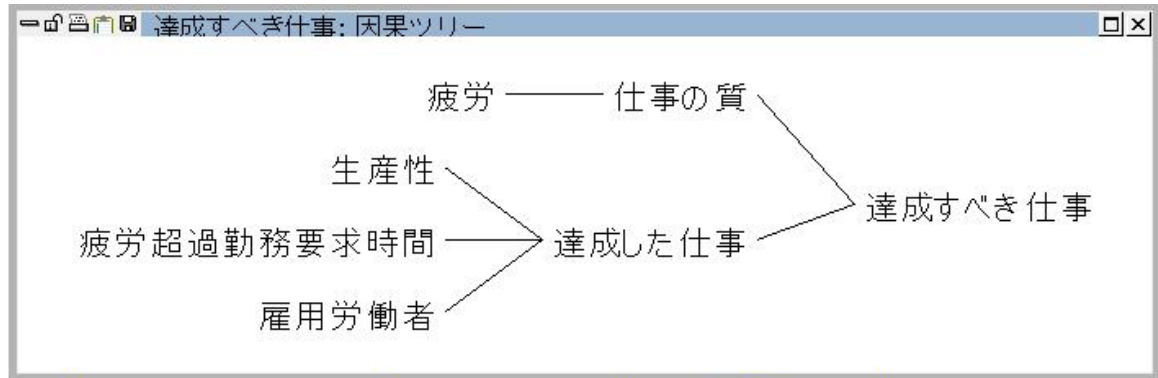
英語版です。


## 4.6 樹形図による因果分析

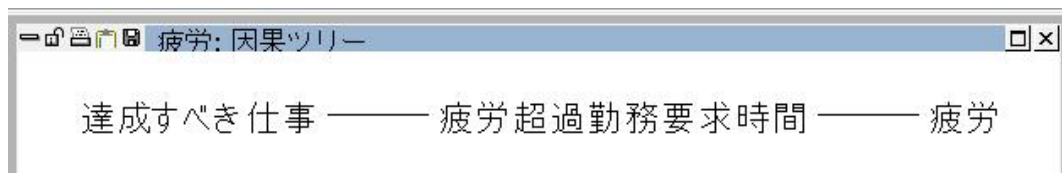
因果分析は、何が変化をもたらしたのかをモデルにおいて突きとめるうえで有効なツールです。因果分析ツールは、変数の因果関係 ( 何が原因となっているのか、またはどのような影響を及ぼすのかの両方 ) を示すことができます。

## 因果ツリー図


- 因果ツリー  を選択します。「達成すべき仕事」の原因が表示されます。

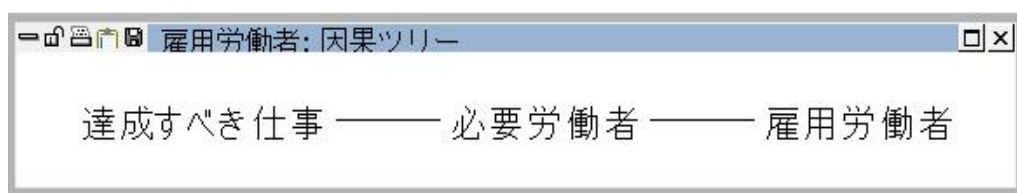


- 樹形図にある「疲労」（文字列）をクリックし、因果ツリー  をクリックします。




「疲労」が「超過勤務要求時間」と「達成すべき仕事」の影響を受けていることがわかります。「達成すべき仕事」から始まり、「達成すべき仕事」までさかのぼるフィードバックループをたどったことになります。次に、「雇用労働者」に影響を及ぼす変数を見ます。

- 樹形図か、もしくはスケッチの「雇用労働者」をクリックし、因果ツリー  をクリックします。





「達成すべき仕事」から「実労働力」を経て、「成すべき仕事」にもどる別の因果ループがあります。

- すでに表示された樹形図の中か、スケッチから「生産性」をクリックし、因果ツリー  をクリックします。




「生産性」には因果関係がなく、このモデルにおいて定数であるか外生変数であることがわかります。




注意 スケッチ画面に戻るには、(1) スケッチ画面をクリックする、(2) 樹形図を削除する、(3) モデルウィンドウ  をクリックする、(4) **[Shift]+[Ctrl]+[Tab]** を使う、という方法があります。ツリー図を削除していなければ、出力ウィンドウ  を繰り返しクリックすることで表示されます。この操作は、**[Ctrl]+[Tab]** でも行うことができます。

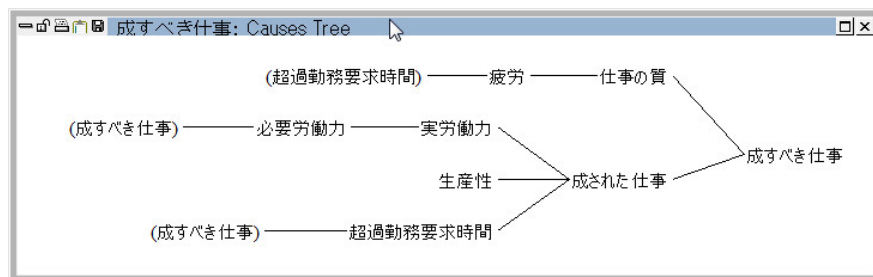
#### 4.6.1 分析ツールオプション (Vensim PLE と PLE Plus は対応しません)

Vensim PLE と PLE Plus のツールセットは変更できません。しかしそれ以外のラインアップでは、分析ツールにさまざまな結果を表示するオプションがあります。そこで、樹形図をより深く表示するように設定します。

- 因果ツリー  を右クリック (または **[Ctrl]** を押しながらクリック) します。樹形図のダイアログオプションが開きます。



- [Tree Diagram Options] の [下層] にあるドロップダウンボックス [ ] をクリックし、[6] を選択し、[OK] をクリックします。
- 「成すべき仕事」をクリックし、因果ツリー  をクリックすることで、「成すべき仕事」に影響する変数が 6 段階先まで表示されます。




「成すべき仕事」と「超過勤務要求時間」は括弧 ( ) で閉じられており、6 段階に到達する前に止まっています。括弧は、この変数が他にも表示されていることを示しています。つまり、この樹形図にフィードバックループが存在することを示しています。

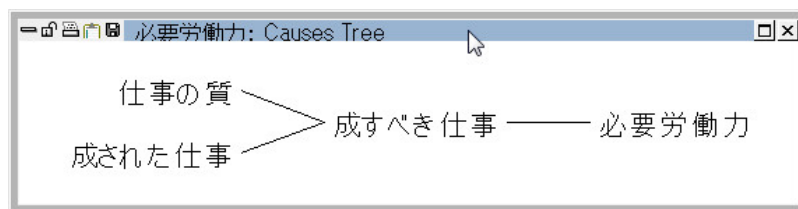



### 自動的なツールの実行

大きなモデルを精査する場合、ある変数を選択（ワークベンチ変数に）した際に、常に分析ツールが自動実行されると効率的です。

- 因果ツリー  を右クリック（または **Ctrl** を押しながらクリック）します。樹形図のダイアログオプションが開きます。
- [Tree Diagram Options] の [下層] にあるドロップダウンボックス [▼] をクリックし、2 を選択します。 初期設定に戻しています。
- [Active on variable selection 【変数選択により実行】] のチェックボックスにチェックを入れ、[OK] をクリックします。
- 「必要労働力」をクリックすると、樹形図が現れます。


因果ツリーが自動的に作成されます。

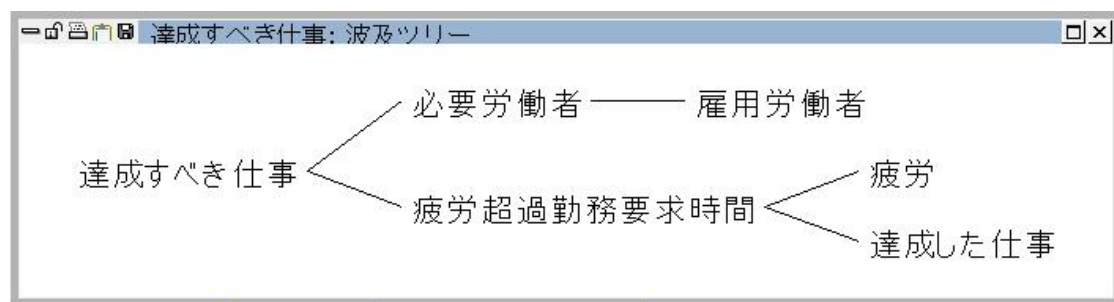


- 因果ツリー  を右クリック（または **Ctrl** を押しながらクリック）します。
- [Active on variable selection] のチェックボックスを外し、[OK] をクリックします。 初期設定に戻しています。

### 波及ツリー図

次に、波及ツリーを説明します。

- 「達成すべき仕事」をクリックし、ワークベンチ変数にします。それはタイトルバーで確認できます。
- 波及ツリー  をクリックします。




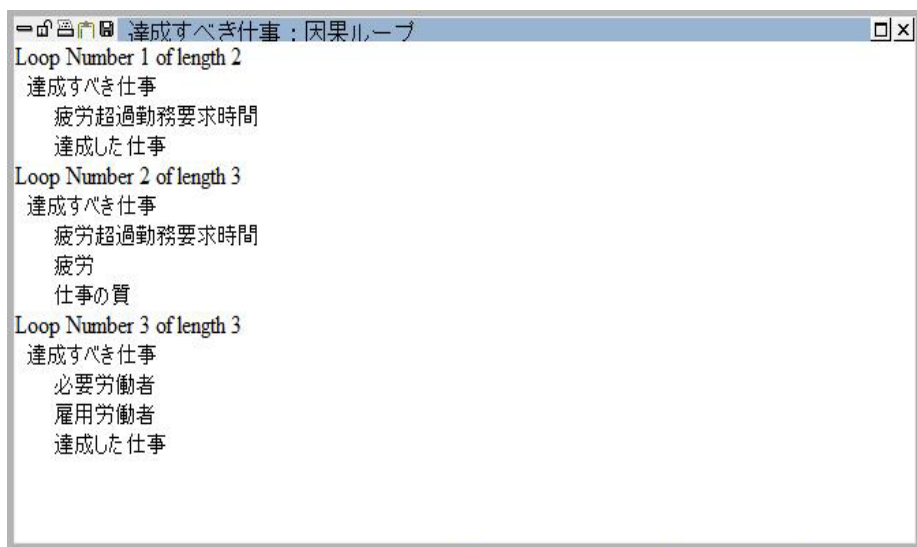
オプションの [Show Link] にある [因果ループ] を利用すれば、「成すべき仕事」がどのような変数から影響を受けているのかが見ることができます。

### ループツール

次に、フィードバックループを探すループツールを紹介します。



ワークベンチ変数でなければ、「達成すべき仕事」をクリックします。




- 「達成すべき仕事」がワークベンチ変数であることをタイトルバーで確認します。
- 因果ループ  をクリックします。「達成すべき仕事」が3つのフィードバックループに関係していることがわかります。

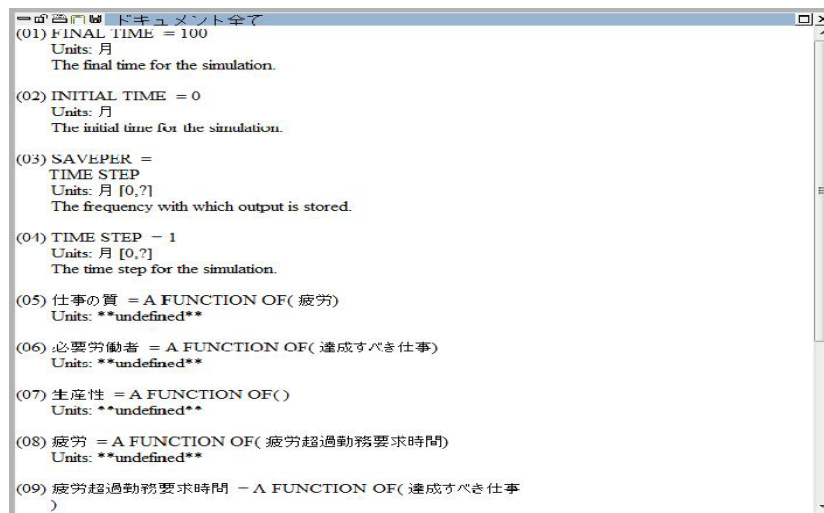


### ドキュメントツール




ドキュメントツールは、モデルをテキスト形式で表示します。Vensim PLE と PLE Plus 以外では、オプションを選択することで、ワークベンチ変数だけを表示したり、すべての変数を表示する設定に変更したりすることができます。初期設定は、ワークベンチ変数の表示のみです。

Vensim PLE と PLE Plus では、ツールバーにドキュメント  とドキュメント  があります。前者はワークベンチ変数だけを表示し、後者は全ドキュメントを表示します。

- Vensim PLE と PLE Plus 以外では、ドキュメント  がツールバーにあれば、それを右クリックし、[Multiple Equation Options 【複数の方程式オプション】] の [全変数] をチェックします。ツールバーが、全ドキュメント  へと変更になります。
- ドキュメント全て  をクリックします。



さまざまなモデルの変数が、アルファベット順に番号を付けられて表示されます。モデルに加えていない *FINAL TIME*、*INITIAL TIME*、*SAVEPER*、*TIMESTEP* という変数が表示されています。これらの4つの変数は、シミュレーションを管理するために利用され、すべてのモデルで存在します。次に、ワークベンチ変数だけを見ます。

- Vensim PLE と PLE Plus 以外では、全ドキュメントツール  を右クリック（または **Ctrl** を押しながらクリック）します。
- [Document Options] の [表示] にある [範囲] と [単位] のチェックボックスをはずします。[Multiple Equation Options] の「切」をクリックします。[OK] をクリックします。アイコンがドキュメント  に切り替わります。
- 「達成すべき仕事」をクリックします。
- ドキュメントツール  をクリックします。



単位の表示がされていません。

ドキュメントの出力では、もしドキュメントウィンドウがすでに存在すれば、それに追加されていきます。もし存在しなければ、新規に作成されます。

#### 分析ツールセットの保存（Vensim PLE と PLE Plus は対応しません）

**注意** Vensim を閉じる時、[Do you want to save the Analysis Toolset] 【このツールセットを保存しますか】というメッセージが表示されます。

- ツールセットを初期設定 (*default.vts*) のままにしたい場合は、[いいえ] をクリックします。
- 変更したツールセットを保存したい場合は、[キャンセル] をクリックし、[ツール] メニュー → [分析ツール] → [別名で保存] を選び、新しいツールセット名を入力し、[保存] をクリックします。





## 第 5 章

# ストックおよびフロー図

ストックおよびフロー図は、因果ループ図で示される情報よりも詳しくシステムの構造を表します。ストックはシステムの振る舞いにとって必須です。またフローはストックに変化をもたらします。

ストックおよびフロー図は、シミュレーションモデルを作成する際に必要となる最初の一步です。なぜならそれらは、因果関係の中心となる変数となるからです。ここでは、製品の認識、さらに顧客と潜在的な顧客の関係についてのモデルを作成します。

### 5.1 作図 ( *customer.mdl* )

- Vensim を起動します。
- [ファイル] メニュー [新規モデル] を選択するか、新規モデル  をクリックします。モデル設定のダイアログに対して、[OK] をクリックします。
- 保存  をクリックし、`\guide\chap05` のディレクトリに、*customer* という名前で保存します。





ファイル名を日本語にすることもできます。

#### 5.1.1 ストックの入力



ストックは蓄積性を持つ変数です。ストックの値は、蓄積および積分の割合によって変化します。これは、たとえフローが非連続的に変化していたとしても、ストックの値が時間とともに連続的に変わることを意味しています。フローは、ストックの値を変化させます。フローの値は、その前の値に依存しません。その代わりに、ストックは、外生の影響も受けつつ、フローの値を決定します。ストックとフローの中間にあたる概念や数式は、補助変数といわれます。この補助変数は、フローのように、ストックあるいは外生の影響により、その値が非連続的に変化することがあります。

ストックおよびフロー図を作成する場合、時間とともにどの変数が蓄積していくのかを考えます。別の方法としては、システムにおいて時間を止めた場合、どの変数に値がまだ存在するかを見ます。例えば、グラスへ水をつぐシステムでは、グラスに注がれた水がストックです。時間を止めた場合、つぐこと（フロー）は止まるでしょう。しかし、まだグラスには水（ストック）があります。まずどのようなストックが必要であるかを明らかにし、次にフローや補助変数を作成します。モデルの作成は、繰り返しのプロセスです。最初から完璧なものを作成しようとはしないでください。あとで修正することは簡単です。

本章であつかうストックは潜在顧客と顧客です。

- ストック変数  によりスケッチをクリックします。「潜在顧客」と入力し、を押します。
- ストック変数  を選択したまま、潜在顧客の右のところをクリックし、「顧客」と入力し、を押します。

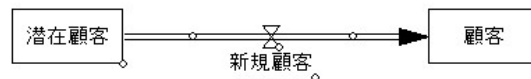
### 5.1.2 フローの作成


- フロー  を選択します。「潜在顧客」を一度クリックします。このとき、マウスボタンをクリックし、放してください。次に、「顧客」にカーソルを移動させて、もう一度クリックしてください。「新規顧客」と入力し、を押します。

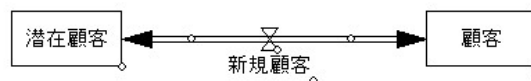
フローには矢印があり、原料が流れる方向を示します。フローはストックを増やすことができる唯一のものです。しかしこの矢印の方向は作図上のことであり、実際には方程式によって原料の方向が決まります。さらに、フローが一方向なのか双方向なのかといったことを作図でも表現することができます。

- スケッチ移動  を選択してください。

パイプの中央には、小さな円があります。




- バルブ  の左側にある小さな円を右クリックしてください。あるいは **[Ctrl] + クリック** します。[ 矢頭 ] のチェックボックスをチェックし、[ OK ] をクリックします。




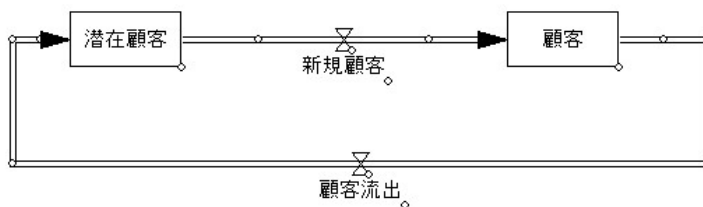
この図は、「新規顧客」が正にも、負にもなることを意味します。このモデルでは、双方向の設定にはしません。そこで次のように元へ戻します。

- [ 編集 ] メニュー [ 元に戻す ] を選択します。または **[Ctrl] + [Z]** を押します。これによって、最後の変更を取り消します。その結果、一方向のフローに戻ります。




フローツールを設定 (PLE と PLE Plus は対応しません) することもできます。例えば、フローを常に双方向として描くには、フロー  を右クリックし、[ One Way Flow ] のチェックを外します。しかし通常、双方向のフローは少ないので、そうした必要が生じた場合に、双方向にするのがよいでしょう。

### 5.1.3 曲がるフローパイプ


- フロー  を用いて、「顧客」を一度クリックして、次に、**[Shift]**を押したまま、「顧客」の右にカーソルを移動させてクリックします。これからの作業は**[Shift]**を押し続けてください。カーソルを下（「顧客」の右下にあたる）に移動し、クリックします。カーソルを「潜在顧客」の左下に移動し、クリックします。カーソルを「潜在顧客」の左に移動し、クリックします。最後に、「潜在顧客」をクリックします。**[Shift]**を放してください。フローの名前を「顧客流出」として、**[Enter]**を押します。

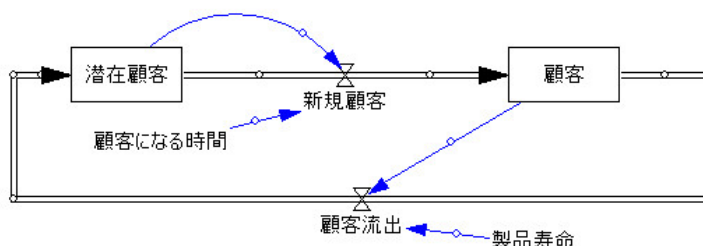


### 5.1.4 補助変数と矢印の追加





- 変数  を選択します。「新規顧客」の左下をクリックし、「顧客になる時間」と入力し、**[Enter]**を押します。
- 「顧客流出」の右下に、「製品寿命」と入力し、**[Enter]**を押します。
- 矢印  を選択します。「潜在顧客」をクリックし、「潜在顧客」と「顧客」の間の余白を一度クリックし、「新規顧客」のバルブ  をクリックします。

曲がった矢印によって、ストックとフローのバルブが連結します。フローのバルブだけでなく、フローの名前を矢印の連結に使うこともできます。フローの名前とバルブは同じ働きをします。




- 「顧客になる時間」をクリックし、フローの「新規顧客」の名前部分をクリックします。
- 「顧客」をクリックし、「顧客流出」のバルブ  をクリックします。
- 「製品寿命」をクリックし、「顧客流出」の名前部分をクリックします。







### 5.1.5 より詳しい構造

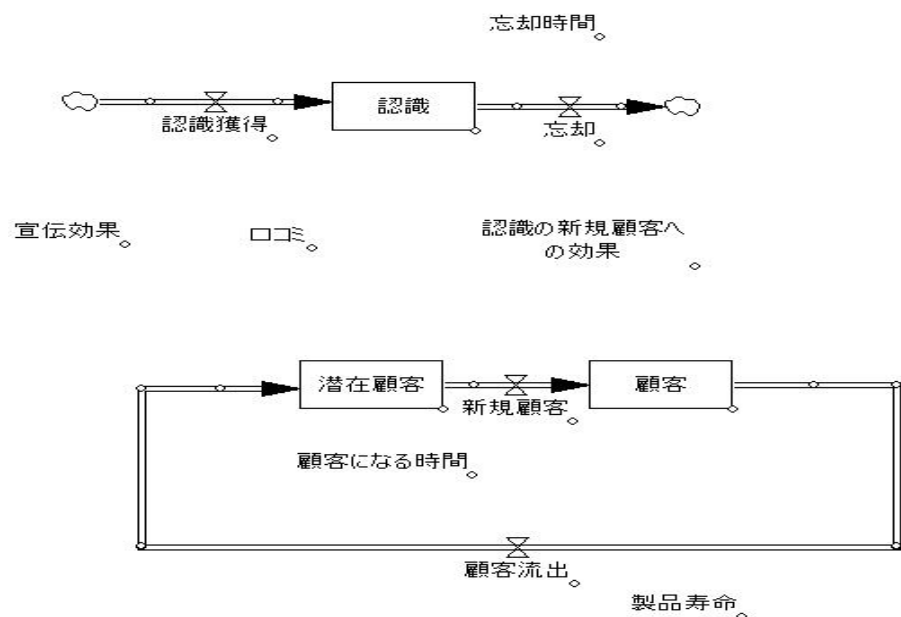
- ストック変数  を選択します。「潜在顧客」の上の余白をクリックし、「認識」を作成し、 をクリックします。
- フロー  を選択します。「認識」の左余白をクリックします。カーソルを「認識」に移動し、このストックをクリックします。名前を「認識獲得」とし、 を押します。

「認識獲得」というフローは、ストックと雲の間に作成されます。雲は、モデルの境界を意味します。この雲からの物質の流入は、具体的にどこから来たのかは対象としません。また雲に流入した物質がどうなかったのかも対象としません。







- ストック「認識」をクリックします。カーソルを「認識」の右に移動させて、クリックします。フローの名前「忘却」を入力し、 を押します。
- 変数  を選択します。「認識獲得」の下をクリックし、「宣伝効果」という名前を入力します。ここで  はしません。編集ボックスを開いたままにします。
- 再度、「認識獲得」の下をクリックし、「ロコミ」という名前を入力します。
- 「忘却」の上をクリックし、「忘却時間」という名前を入力します。

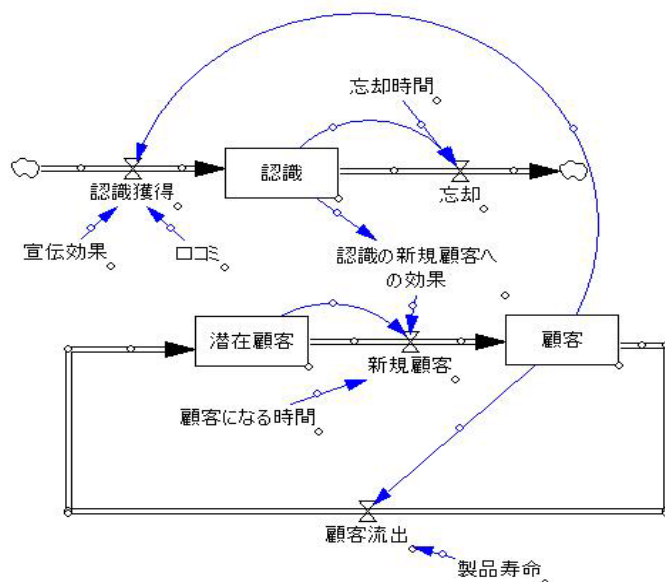
 を押さないで、続けてスケッチをクリックすることで、多数の変数を入力することができます。

- 「認識」と「新規顧客」の間をクリックし、「認識の新規顧客への効果」という名前を入力し、 を押します。
- もし作図スペースを使い果たしていれば、図を移動させることができます。[編集]メニュー→[全て選択]を選びます。または **Ctrl** + **A** を選択します。すべての図が選択され、それらを移動することができます。このとき、スケッチ固定  が選択されていると、動かせません。スケッチ移動  を選択します。選択されている図の外側をクリックすると、選択が解除されます。







- 矢印  を選択します。「宣伝効果」をクリックし、「認識獲得」をクリックします。
- 「顧客の紹介」をクリックし、「認識獲得」をクリックします。
- 「認識」をクリックし、その右上の余白をクリックし、「忘却」のバルブ  をクリックします。
- 「忘却時間」をクリックし、「忘却」のバルブ  をクリックします。
- 「認識」をクリックし、「認識の新規顧客への効果」をクリックします。
- 「認識の新規顧客への効果」をクリックし、「新規顧客」のバルブ  をクリックします。
- 「顧客」をクリックし、フロー「忘却」の雲周辺の余白をクリックし、「認識獲得」のバルブ  をクリックします。
- 変数の場所を動かしたり、矢印の形状を変えるためのハンドル操作を行うためには、スケッチ移動  を選択します。これによってスケッチの見栄えを良くします。




- 保存  をクリックし、モデルを保存します。

## 5.2 図のカスタマイズ


図形は様々な方法でカスタマイズすることができます。Vensim における標準的な技法では、箱の内部に名前のある変数はストックです。フローは、名前を持つバルブ  です。フローはバルブ  だけのこともあります。補助変数や定数、表関数、データ（外生）変数などは、名前だけを表示します。本節では、図のカスタマイズを行うためのオプションを紹介します。

### 5.2.1 スケッチオプション

レイアウトメニューでは、最後に選択された対象を基準に実行されます。したがってこの場合、「認識」のサイズにあわせることになります。

- スケッチ移動  を選択します。ストック「認識」にあるハンドル（箱右下の小さい円）をクリックし、箱が少し大きくなるようにドラッグします。
- スtock「潜在顧客」をクリックし、**[Shift]**を押したまま、「顧客」、「認識」をクリックします。**[レイアウト]**メニュー **[大きさを揃える]**を選択します。
- **[編集]**メニュー **[全て選択]**を選びます。これは**[Ctrl]+[A]**でも可能です。ステータスバーにある矢印の配色ボタンをクリックします。このボタンを見つけるには、カーソルをボタン上に動かし、しばらく放置すると「選択した矢印の色をセットする」と表示されるものを選びます。色（例えば赤）を選択し、選択された箱の外側をクリックします。

矢印とフローの両方が赤に変更されることに注意してください。矢印とフローを別々に変更したければ、それぞれを別を選択する必要があります。それをメニューで行うには、**[編集]**メニュー **[一部選択]** **[情報の矢印]**や、**[編集]**メニュー **[一部選択]** **[フロー]**を用います。また、**[Shift]**を押したまま、矢印（もしくはフロー）を個別にクリックする方法もあります。





- 「認識」をクリックし、**[Shift]**を押したまま、「潜在顧客」と「顧客」をクリックします。ステータスバーの**b**をクリックし、太字とします。このボタンは、**B**と切り替わります。ステータスバーには、太字以外にも、変数の色や箱の色も変更するボタンがあります。
- **[Esc]**もしくは、**[1]**またはスケッチ固定  をクリックし、スケッチを固定します。

### 5.2.2 変数の形状

マウスの右ボタンで変数をクリックし、オプションダイアログで新しい形状を選択することができます。

- 「製品寿命」を右クリックします。
- **[形状]**において**[円]**のオプションボタンをクリックし、**[文字の位置]**を**[下]**とします。**[OK]**をクリックします。



これと同じことは、次の操作でも可能です。スケッチ固定  かスケッチ移動  を選択し、この変数を強調表示します。そのうえで、ステータスバーの選択した変数のボックスの形をセットする  や選択した変数のテキストの位置をセットする  といったボタンを利用します。

### 5.2.3 デフォルトのスケッチオプション (PLE と PLE Plus は対応しません)



[ ツール ] メニュー [ オプション ] で表示されるダイアログには、変数の型ごとの形状が選択できます。

➤ [ ツール ] メニュー [ オプション ] を選択し、[ スケッチ ] タブをクリックします。




最も下にある [ By Type Shapes ] 【型ごとの形状】には、変数の型ごとにドロップダウンボックス [ ] があることに注意してください。

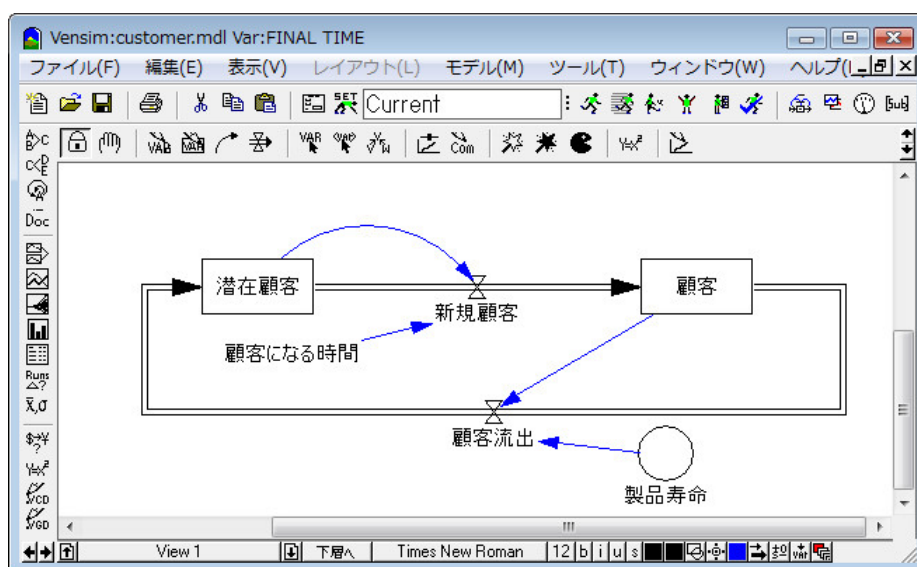
➤ [ キャンセル ] をクリックします。


変数の型ごとの形状を設定する別の方法には、スケッチ固定  により変数を選択し、ステータスバーにある選択した変数のテキストの位置をセットする  をクリックして表示される形状を選択します。

#### 5.2.4 スケッチ上の要素を隠す (PLE と PLE Plus は対応しません)

図をカスタマイズし、さらに [表示] メニューの機能を見ます。


- 下層に隠す  を選択します。「認識」、「忘却」、「認識を得る」、「忘却時間」、「認識の新規顧客への効果」、「宣伝効果」、「顧客の紹介」をクリックします。また2つの雲もクリックします。





- [表示] メニュー [下層表示] [下層1] を選択します。または  をクリックします。

ステータスバーの [下層へ] というボタン表示が、[下層1] へと変わります。



[表示] メニュー [下層表示] で、何番目の層が表示されているのかを確認できます。下層を選択することで、下層に隠す  により表示されなかった要素が表示されます。

- 上層に戻る  を用いて、隠されていた要素を1つずつクリックします。
- [表示] メニュー [下層表示] [なし] を選択します。または  をクリックします。

スケッチされた要素はすべて表示されているはずです。もし何かを見逃していれば、それは隠されたままになります。それを表示するには、上記のプロセスを繰り返します。

注意 層は 16 段階セットすることができます。層は ☐ や ☐ をつかって、連続的に表示する層を設定することができます。

### 5.2.5 画面表示の変更

➤ [表示]メニュー [拡大] [200%] を選択します。

スケッチの表示は 200% になります。

➤ [表示]メニュー [拡大] [100%] を選択します。

スケッチは元のサイズに戻ります。[拡大] は単に、スケッチを拡大したり縮小するものです。これと同じ機能をマウスを使って行うこともできます。そのためには、**[Shift]+[Ctrl]** の両方をクリックし、マウスをスケッチ上で上下にドラッグします。

#### 画面表示の縮尺変更

➤ [表示]メニュー [再スケール] を選択します。[水平] の編集ボックスには「120」を、[垂直] には「140」を入力し、[OK] をクリックします。


縮尺変更は、スケッチされているオブジェクト間の大きさや距離を変更します。縮尺変更はそれ以外の条件は変更しません。注意すべきことは、100% と入力しても、元のサイズには戻せないということです。つまり、縮尺を与える必要があり、上記の例では、X 軸に 83%、Y 軸に 71% とする必要があります。

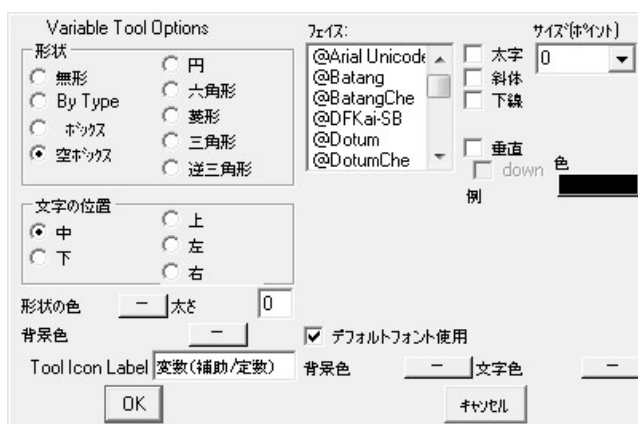
$100/120=83\%$ 、


$100/140=71\%$  のため。





## 5.3 スケッチツールの修正 (PLE と PLE Plus は対応しません)

これまでの作図では、デフォルトのスケッチツールを使用してきました。これらのツールを個別に修正することができます。また、使用しているツールをセットとしても変更することができます (詳細は、リファレンスマニュアルの 13 章を参照)。スケッチツールの中で最も多い修正は変数の形状です。

➤ 変数  を右ボタンでクリックします。次のようなダイアログが表示されます。



ストック変数  では「形状」が「ボックス」になっています。

このダイアログは、変数オプションとほとんど同じですが、若干の追加があります。  
変数  とストック変数  における唯一の違いは、形状の選択です。変数  では、「形状」が「空ボックス」です。この変数  は、ハンドルを持つ空ボックスを作成します。ハンドルを操作することで、長い名前の変数を複数行で表示することができます。  
もし変数を一行で表示する必要がある場合は、「形状」を「無形」とします。

➤ ダイアログを閉じるために、「OK」をクリックします。

### 5.3.1 スケッチツールセットの保存

**注意** Vensim を終了する際に、「Do you want to save the current toolset」【現在のツールセットを保存しますか】というメッセージが表示されることがあります。この場合、「はい」「いいえ」「キャンセル」が選択できます。

➤ 「いいえ」を選択すると、デフォルトのスケッチツールセット (default.sts) の設定が残されます。

新しいセットを保存する場合には、「キャンセル」をクリックし、「ツール」メニュー「分析ツール」 「別名で保存」を選択し、新しいツールセット名を入力後、「保存」をクリックします。

## 第 6 章

# シミュレーションモデルの構築

### 6.1 うさぎ頭数モデル

この章では、うさぎ頭数のシミュレーションモデルを紹介します。モデルの作成は、まずモデルをスケッチすることから始めます。次に、方程式を書き入れ、数量化します。モデルを実行すると、自動的にデータセットという出力が保存されます。最後に、モデルにおける変数のダイナミックな振る舞いを、分析ツールでチェックします。

通常、モデル作成は次のような方法をとります。つまりモデルの作成、検討、修正です。これを繰り返して、モデルを改良します。デバッグ（モデルの正確な作成）とモデル分析（出力された振る舞いを調べること）によって、モデルは改良されます。リアリティチェックはモデルの作成と改良を助けるもう 1 つの方法です。これについては、14 章で紹介します。

シミュレーションモデルにおける振る舞いは、さまざまな変数間の関係を決定する方程式によって決まります。このガイドでは、シミュレーションモデルの方程式をすべて明示します。モデル図（因果ループ図やストックおよびフロー図）は、変数の関係を図で示します。Vensim は、図と方程式の整合性を持ちますが、図の中で情報が省略されたり、非表示になることもあります。この章でのシミュレーションモデルの作成では、方程式がこのマニュアルに書かれているものと一致することを確認してください。もし、図が違う場合は、変数を隠しているか、もしくはモデル図の表示に影響を及ぼすオプションが選択されている可能性があります。

### 6.2 Vensim の表記法

#### 名前の付け方








モデルの作成、分析、プレゼンテーションをわかりやすくするために、モデルで使用する名前は工夫が必要です。このマニュアルにあるモデルは、推奨される表記法に沿っています。

ストックは、初めの文字が大文字で表示されます。（例えば *Population* 【人口】）


フロー、補助変数、定数、表関数、データ変数などは、すべて小文字で表示されます。例えば、（*average lifetime* 【平均寿命】）です。


日本語ではこうした表記はできません。しかし Vensim における豊富なサンプルモデルをみる場合、こうした表記法を知っておくと理解が容易になります。

### スケッチ



ストックは、ストック変数  で入力します。ストック変数  を使って作成された変数は、ストックとみなされます。方程式  で方程式編集ダイアログを開いた時、その変数がストック変数として作成されていることが [タイプ] が [ストック] であることからわかります。ここで、変数の [タイプ] を変えることもできます。また、変数  で作成された (箱を持たない) 変数をストック変数にすることもできます。しかしこれは混乱しやすいため、特別な目的以外は、勧めません。フローは、フロー  で入力します。デフォルトの設定では、フローを作成した方向に名前を持つ矢印が描かれます。フローの名前を削除するには、フローの編集ボックスが表示された際に、**[Esc]** を押します。フローのもう 1 つの端に矢印を付けることもできます。スケッチ移動  を選択し、矢印を付ける側のハンドルを右クリックし、[矢頭] をチェックします。PLE と PLE Plus 以外では、フロー  を右クリックし、[One Way Flow] 【一方向のフロー】のチェックを外すことによって双方向のフローが作成されます。また [Query Valve Name] 【バルブ名の問い合わせ】のチェックを外すことによって、名前を持たないフローが作成されます。

**重要な注意** フローにおける矢印があってもなくても、シミュレーションモデルにおけるフローの方程式には影響しません。例えば、フローの矢印があるストックを増加させることを示していても、方程式でストックを減らすことが可能です。フローを正確に示すのは、その方程式であって、図ではありません。

定数、補助変数 (表関数、データ、さらに他の変数) などは、変数  で作成します。それらは箱のない文字列で通常入力します。つまり、その変数のオプションは、[形状] が [空ボックス] または [1 行表示] となっています。補助変数や定数を円形 (その下に名前を付ける) で表現することもあります。この場合、図が乱雑になることもあります。変数の型ごとに異なる形状を設定すれば、図から伝えられる情報は増えます。しかし標準的な図の表記法は決まっていません。また多くの図の形状を設定すると、視覚的に混乱します。したがってシミュレーションモデルの作成では、ストックを箱で、フローはバルブで、それ以外の変数は文字列だけで表現することを勧めます。









PLE や PLE Plus 以外では、スケッチツールのオプションを設定することができます。例えば、変数  で、[形状] が [円] や [六角形] などを作成するようにすることができます。

## 6.3 うさぎ頭数モデルの作成 (pop.mdl)

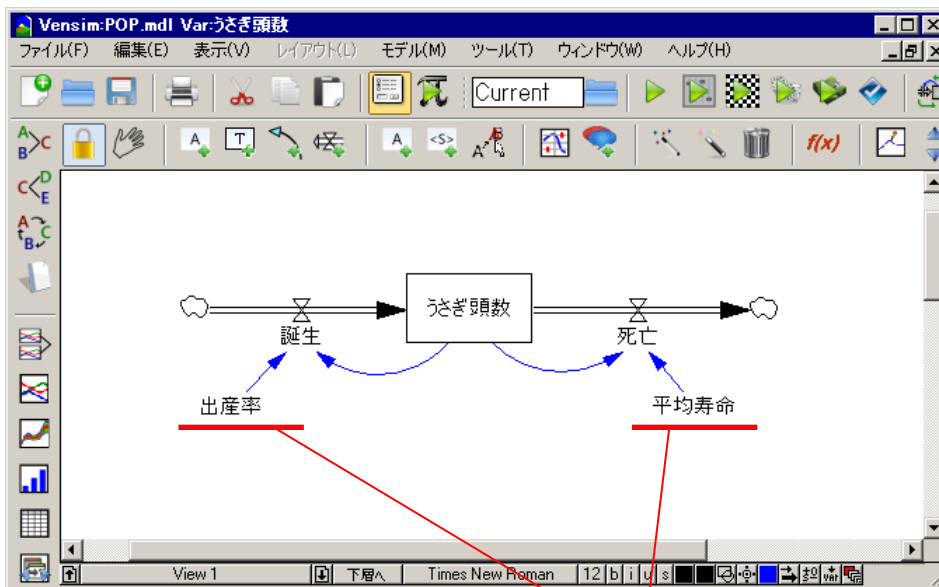
- 新規モデル  をクリックするか、[ファイル] メニュー [新規モデル] を選びます。
- モデル設定ダイアログの [時間の範囲] タブから、[終了時間] を「30」と入力します。[時間ステップ] を「0.125」と入力するか、ドロップダウンボックス [ ] をクリックし、0.125 を選びます。続いて [時間単位] のドロップダウンボックス [ ] をクリックし、[年] を選び [OK] をクリックします。**[↵]** を押します。
- スtock変数  をクリックします。スケッチの中央あたりをクリックし、名前を「うさぎ頭数」とし、**[↵]** を押します。

年がなければ、直接入力します。



- フロー  をクリックします。「うさぎ頭数」の左で一度クリックし、カーソルを動かして、「うさぎ頭数」をクリックします。名前を「誕生」とし、を押します。
- 「うさぎ頭数」をクリックし、カーソルを右方向に動かして、もう一度クリックします。名前を「死亡」とし、を押します。
- 変数  をクリックします。「誕生」の左下をクリックし、名前を「出生率」とし、を押します。「死亡」の右下をクリックし、名前を「平均寿命」とし、を押します。
- 矢印  をクリックします。出生率をクリックし、次に「誕生」をクリックします。「平均寿命」をクリックし、次に「死亡」をクリックします。
- 「うさぎ頭数」をクリックし、次に「うさぎ頭数」のほんの少し左下をクリックし、「誕生」をクリックします。
- 「うさぎ頭数」をクリックし、次に「うさぎ頭数」の右下をクリックし、「死亡」をクリックします。
- 保存  をクリックします。このモデルを *guide\chap06* に保存します。名前を *pop.mdl* とします。

うさぎ頭数モデルの構造が下の図のように作図されました。「うさぎ頭数」から「誕生」への正のフォードバックループが「うさぎ頭数」を増加させ、「死亡」からの負のフィードバックループが「うさぎ頭数」を減少させています。



補助変数

## 6.4 方程式の入力

うさぎ頭数モデルの構造は完成しました。しかし、このままではシミュレーションは実行できません。シミュレーションを実行するには、それぞれの関係を示す方程式を設定しなければなりません。これらの方程式は、簡単な代数表現です。つまりある変数を、それに影響する別の変数で示します。例えば、

誕生 = うさぎ頭数 \* 出生率

スケッチを見ると、「出生率」には因果関係がなく、値が定数であることがわかります。定数は数値を持ちます。

入力した方程式は、単位を持ちます。方程式における単位の整合性をチェックするために、単位は用いられます。単位の整合性は、モデル構造を修正するうえで重要です。そのためには、[モデル] メニュー→[単位チェック] を選ぶか、**Ctrl**+**U** を用います。

➤ 方程式 **f(x)** をクリックします。

すべての変数が反転表示されます。強調表示は、完成度を目に見える方法でチェックします。強調表示されているところは、変数に方程式が入力されていないか、方程式に不備があることを示しています。方程式がすべて入力されれば、強調表示はなくなります。[モデル] メニュー→[モデルチェック] を選ぶか、または、方程式 **f(x)** の方程式編集ダイアログから、[モデルチェック] を選ぶと、不備のあるところが表示されます。

➤ 「誕生」をクリックします。

方程式編集ダイアログが開きます。表示される内容は、使用している Vensim のラインアップによって異なります。Vensim PLE と Vensim PLE Plus は簡略化されていますが、主な構成は同じです。

#### 変数の種類：補助変数

方程式編集ボックスの上の欄には、クリックした変数の名前「誕生」が表示されています。左の[タイプ]は、[補助変数]となっています。Vensim は、フロー＝補助変数は同じ変数の種類とみなします。ドロップダウンボックス [▼] をクリックして、他の種類を見えます。リストを確認し、まだ[補助変数]が選択されていることを確認します。次に、方程式編集ボックス(=の右横)をクリックします。

- 誕生の方程式を次のように完成させます。
- 「うさぎ頭数 \* 誕生」と入力します。もしくは、
- 方程式編集ダイアログの中央にある変数リストのなかから、「うさぎ頭数」をクリックし、次に[\*]を入力（または、方程式編集ダイアログの中央にあるキーパッド[\*]をクリック）します。さらに変数リストのなかから、「出産率」をクリックします。

方程式をわかりやすくするために、スペースや改行を加えることもできます。改行は、**[Ctrl]+[Enter]**を押します。変数の名前の途中で改行することはできません。次に、「誕生」の単位を入力します。

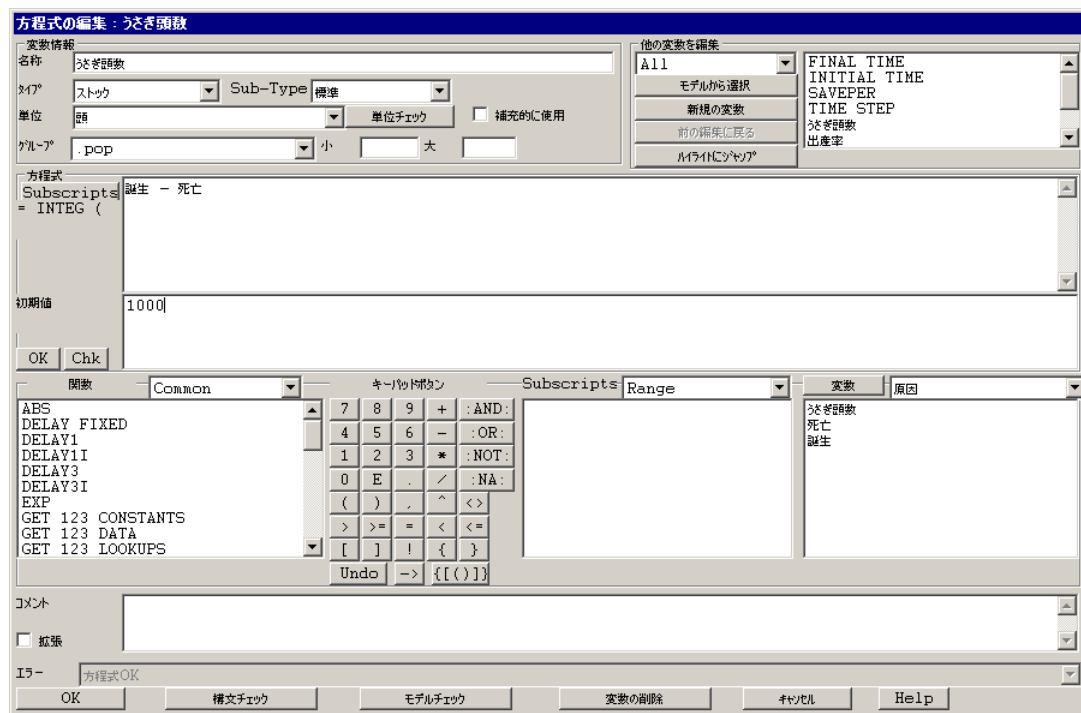
- [単位] に「頭/年」と入力します。1 年間に産まれるうさぎの数を単位とすることを表しています。**[OK]** をクリックするか、**[Enter]** を押します。

モデル構造や方程式に問題がなければ、ダイアログが閉じられます。もし問題があった場合は、間違いを指摘するエラーメッセージが表示されます。

#### 変数の種類：ストック

- 「うさぎ頭数」をクリックします。

方程式編集ダイアログが開きます。先ほどの「誕生」とは多少異なります。





左の「タイプ」では、変数の種類が「ストック」になっています。方程式編集ボックスは、INTEG 関数がストックを定義しています。INTEG は時間とともに変数を integration【積分】するという意味です。「誕生」と「死亡」が、すでにストックに関連づけられている（フローが入力されている）ため、方程式編集ボックスには方程式が表示されています。Vensim は自動的にストックの方程式に、フローの変数を入力します。ストックに流入するフローは正（プラス）で、流出するフローは負（マイナス）として扱われます。もしフローの向きが違う場合には、方程式編集ボックスで符号を変えることができます。このモデルでは、「誕生」は「うさぎ頭数」を増やし、「死亡」は「うさぎ頭数」を減らすので符号の修正は入りません。

この方程式編集ダイアログには、「初期値」を設定するための編集ボックスがあります。カーソルをそちらに移動します。

- 初期値には「1000」を入力します。

この値は、シミュレーション開始時のうさぎの頭数です。

- 「単位」に「頭」と入力し、[OK] をクリックするか、を押します。
- 「出産率」をクリックし、編集ボックスに「0.125」と入力します。
- 「単位」に「1/年」と入力します。これは出産率が1年間における出産頭数とするためです。これをわかりやすく説明します。出産率は“（うさぎの誕生数/総うさぎ頭数）/年”を意味しています。この単位は、“（頭数/頭数）/年”であり、頭数が約分されて“1/年”となります。  
[OK] をクリックするか、を押します。
- 残る2つの方程式についても、下記のリストを参照し、完成させます。

モデルの変数は、すべての方程式が入力されると、反転表示で強調された変数がなくなります。また一度入力された単位は、ドロップダウンボックス [▼] から選択することができます。

pop.mdl の方程式

```

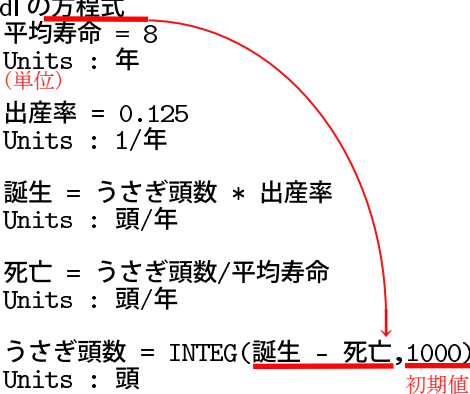
平均寿命 = 8
Units : 年
(単位)
出産率 = 0.125
Units : 1/年

誕生 = うさぎ頭数 * 出産率
Units : 頭/年

死亡 = うさぎ頭数/平均寿命
Units : 頭/年

うさぎ頭数 = INTEG(誕生 - 死亡, 1000)
Units : 頭
初期値

```




## 6.5 構文と単位のチェック

モデルをシミュレーションする前に、方程式と単位をチェックします。

- [モデル] メニュー→ [モデルチェック] を選びます。または **Ctrl**+**T** を押します。[モデル O.K. です] と表示されます。間違いが見つかった場合、その変数の編集ダイアログが現れます。入力した方程式がリストと同じかどうか確認してください。モデルの構造も、上記の図と同じかどうか確認してください。
- [モデル] メニュー→ [単位チェック] を選びます。または **Ctrl**+**U** を押します。[単位 O.K. です] と表示されます。

エラーが表示された場合、どの変数に間違いがあるのかを確認します。問題のある変数の方程式編集ダイアログを開いて、先ほどのリストと違う部分や単位をチェックします。単位に問題がある場合は、構文に間違いがあることが考えられます。




**注意** 単位チェック  は、分析ツールにもあります (Vensim PLE と PLE Plus は対応しません)。

### 6.5.1 単位の同意性

単位を入力する際、英語の場合、単数形と複数形のどちらも使用することがあります。また日本語の単位でも、「頭」と「頭数」が混在する場合も考えられます。本来なら同じ単位であっても、Vensim はその違いを区別することができません。これを解消するには、単位を同じものに統一します。または次のように、同意性を定義することもできます。

- [モデル] メニュー→ [設定] を選び、[単位同義語] タブをクリックします。編集ボックスに、「頭, 頭数」と入力し、[編集して追加] をクリックします。[OK] して、編集ボックスを閉じます。

### 6.5.2 モデルのシミュレーション

- ツールバーにあるシミュレーション名の編集ボックス  をダブルクリックし、「均衡」と入力します。
- シミュレーションの実行  をクリックします。または、カーソルがシミュレーション名の編集ボックスにある時に、 を押します。



モデルがシミュレーションされます。このモデルは小規模なため、何も変化がないようにみえます。実行されているかを確認するには、再度シミュレーションを実行し、「データセットを上書きしますか?」という表示が現れたら、モデルはシミュレーションされていることを示しています。

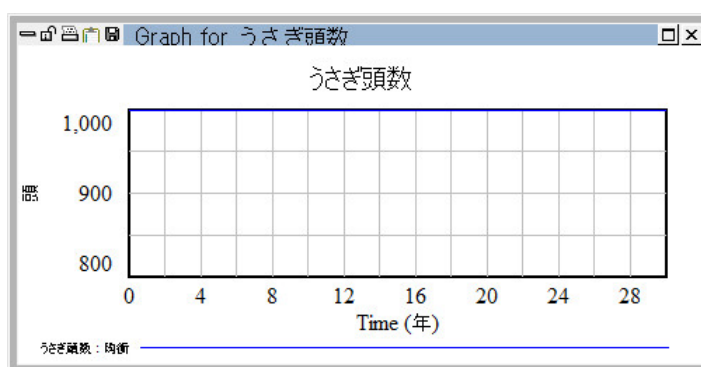
## 6.6 モデルの分析

このモデルは、うさぎの頭数が、均衡を保つように作成されています。定数の「出産率」と「平均寿命」が、「うさぎ頭数」の 12.5% のフローを生み出すように設定されています。したがって、それぞれのフィードバックループは量的な均衡が保たれています。その結果、「うさぎ頭数」には変化が現れません。


0.125=1/8

### グラフと表ツール

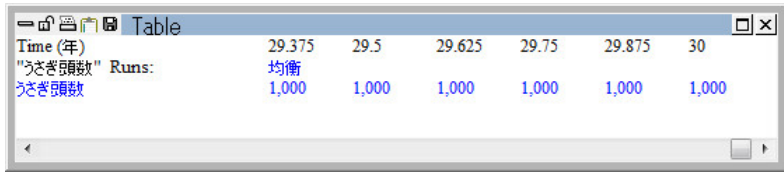
- スケッチの「うさぎ頭数」をクリックします。  
このクリックで、「うさぎ頭数」がワークベンチ変数となります。この他に、制御パネル  の変数タブから、「うさぎ頭数」を選ぶ方法もあります。タイトルバーに「うさぎ頭数」が表示されていることを確認してください。
- グラフ  をクリックします。「うさぎ頭数」のグラフが表示されます。



「うさぎ頭数」がグラフの一番上の、1000 頭のところに直線で示されており、変化がないことがわかります。

- 表  をクリックします。

出力ウィンドウは、「うさぎ頭数」に変化がないことを示しています。スクロールバーを動かして、値が常に 1000 であることを確認します。



Time (年)	29.375	29.5	29.625	29.75	29.875	30
"うさぎ頭数" Runs:	均衡					
うさぎ頭数	1,000	1,000	1,000	1,000	1,000	1,000


## 6.7 シミュレーションの比較

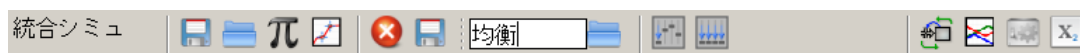
Vensim の際だった特徴は、条件を変えて実行したシミュレーションを比較できることにあります。また実行したシミュレーションごとに、すべての変数のデータを蓄積しています。そのため、どの条件で、どの変数が、どのような振る舞いをしたのかを容易に知ることができます。シミュレーション実験は、定数や表関数の値を、一時的に変えることで行われます。したがって基本的なモデル構造は同じのままです。

### 6.7.1 指数的な増加

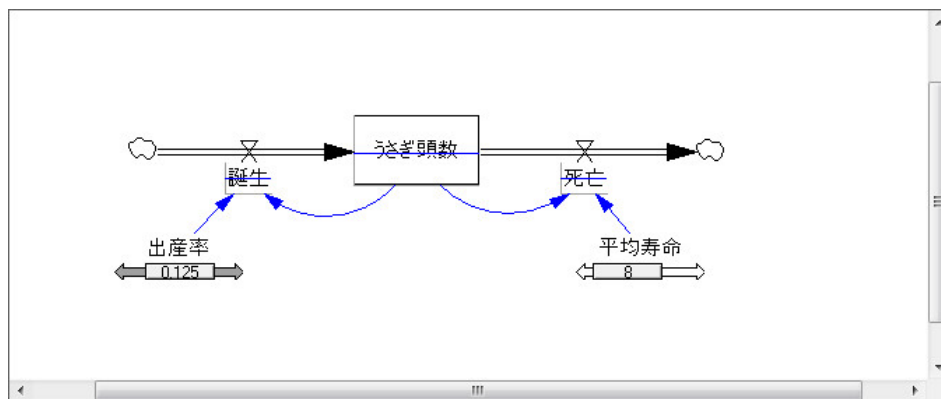
これまでは、均衡状態のシミュレーションでした。次にモデルの定数を変化させて、一定ではない成長を発生させます。これは指数的な増加と言われるダイナミックなものです。

### 6.7.2 シミュレーション実験

- 統合シミュレーション  をクリックし、データセットを上書きします。ツールバーが変わります。

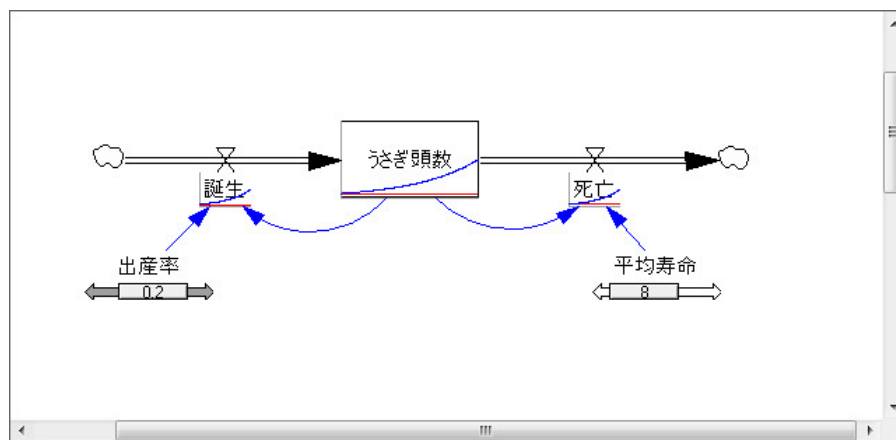


図の中の変数が以下ようになります。



- シミュレーション名を「均衡」から「成長」に変えます。
- 変数「出生率」の下のスライダを動かし、値を「0.2」にします。マウスを使ってもうまいかな場合は、キーボードの矢印キーを使って、値を変更します。スケッチ

は以下ようになります。



青いラインは実行中の結果を、赤いラインは「均衡」の結果を表しています。

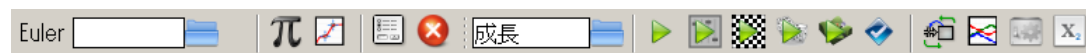
- 停止 をクリックし、統合シミュレーションをストップします。  
スケッチからグラフとスライダが消えます。
- **[B]**を押すか、[表示] メニュー → [グラフ簡素表示] を選び、再度グラフを表示させます。**[B]**を押すことで、グラフの表示、非表示を切り替えることができます。

### 6.7.3 シミュレーションのセットアップによる方法

統合シミュレーション以外の選択肢として、シミュレーションのセットアップと実行を同時に行うこともできます。シミュレーションに数秒以上かかるような、大きいモデルの場合、統合シミュレーションよりも実用的です。

- シミュレーションの準備 をクリックします。

ツールバーがシミュレーションツールバーに変わります。





このツールバーは、セットアップによるシミュレーション用のものです。スケッチの定数が、青い背景に黄色の文字に変わっています。また、スケッチツールが薄く表示され、スケッチで利用できなくなります。

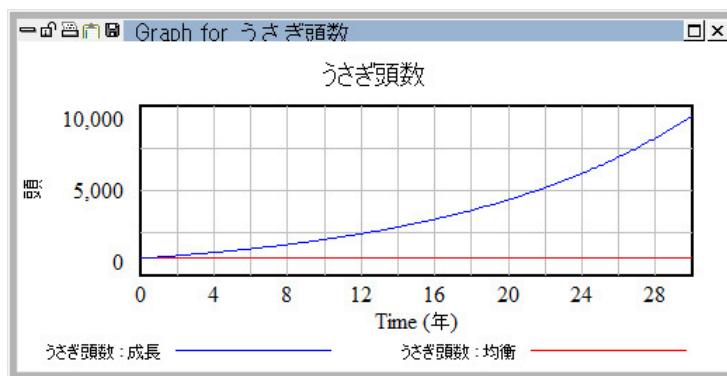
- シミュレーション名を「均衡」から「成長」に変更します。
- スケッチ上で黄色/青で表示された変数「出産率」をクリックし、編集ボックスに「0.2」と入力し、を押します。これは、一時的な値であって、実行後は元に戻ります。
- シミュレーションボタン をクリックし、シミュレーションを実行します。統合シミュレーションモードで、すでに「成長」というシミュレーションを作成している場合は、[データセット成長はすでに存在します。上書きしますか?] と表示されるので、[はい] をクリックします。




- [B]を押すか、[表示]メニュー [グラフ簡素表示]を選び、グラフの動きを表示させます。

#### 6.7.4 直接原因グラフ

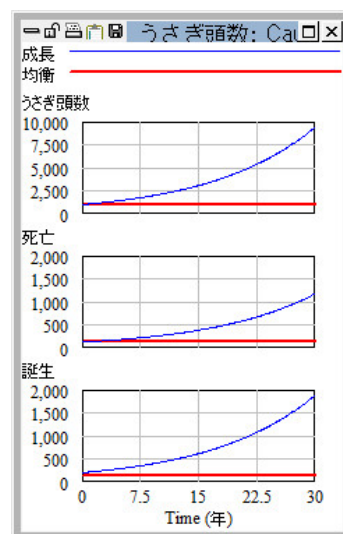
- 「うさぎ頭数」をクリックします。
- 制御パネル  をクリックし、コントロールパネルを前面に表示させます。[データセット]タブをクリックし、2つの実行名が右の欄に書き込まれていることを確認します。  
最後に作成したシミュレーション名(成長)が、上位に表示されています。ほとんどの分析ツールは、両方のデータセットに対して働き、実行結果を比較します。
- グラフ  をクリックします。



2つの実行結果が表示されています。


- [Del]をクリックするか、閉じるボタンでグラフを閉じます。
- 直接原因グラフ  をクリックします。

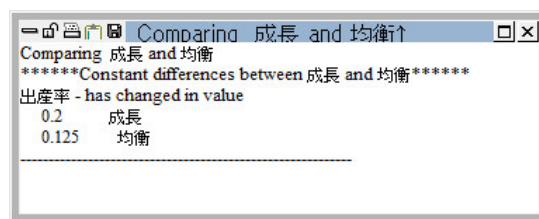
「うさぎ頭数」と、その原因となる「誕生」と「死亡」のグラフが作成されます。



### 6.7.5 シミュレーションの比較

初めのシミュレーションと 2 つ目のシミュレーションとの違いを調べるために、すべての定数（及び表関数）の違いを比較するツールがあります。このツールは最後に作成されたものと、その前に作成されたものを比較します（制御パネルの [ データセット ] にある上位 2 つのシミュレーション結果に対応します）。

- 比較の実行  をクリックします。「均衡」と「成長」における定数「出生率」の違いが、以下のように表示されます。





「うさぎ頭数」は、シミュレーション「成長」では増加します。なぜなら、「出生率」が高く設定されたからです。これによって、「死亡」に関連する負のフィードバックループより、「誕生」に関連する正のフィードバックループが強力になり、その結果、「うさぎ頭数」が時間とともに増大します。

- [ ウィンドウ ] メニュー [ 全出力ウィンドウを閉じる ] を選びます。  
分析ツールを使って作成したすべてのウィンドウが閉じられます。




### 6.7.6 指数的な減少

次に、「うさぎ頭数」が指数的に減少するモデルを作成します。そのために、定数を変化させます。



#### 第 1 の方法

- シミュレーション名をダブルクリックし、名前を「成長」から「減少」に変えます。
- 統合シミュレーション  をクリックします。
- 「平均寿命」の下のスライダを動かして、値を「4」にします。
- 停止  をクリックします。


#### 第 2 の方法

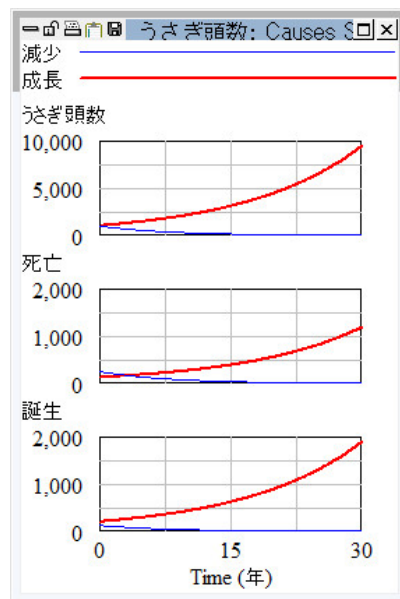
- シミュレーションの準備  をクリックします。
- スケッチ上で黄色/青で表示された平均寿命をクリックし、編集ボックスの値を「4」と入力します。 を押します。
- シミュレーションの実行  をクリックします。

いずれも、新しいシミュレーション「減少」を作成する方法です。

- グラフ  をクリックし、3 つのシミュレーションを比較してみます。
- メインツールバーの制御パネル  をクリックします。[ データセット ] タブの右の欄にある、「均衡」をダブルクリックします。これによって、「均衡」は取り除か

れます。したがって分析ツールの対象から外れます。再読み込みするには、左側にあるデータセット名をクリックし、移動 >> ボタンで右側に移動させます。


- 直接原因グラフ  をクリックします。「うさぎ頭数」と、それに影響を与える「誕生」と「死亡」の一連のグラフが作成されます。ただし表示されるグラフは「成長」と「減少」だけであることに注意してください。

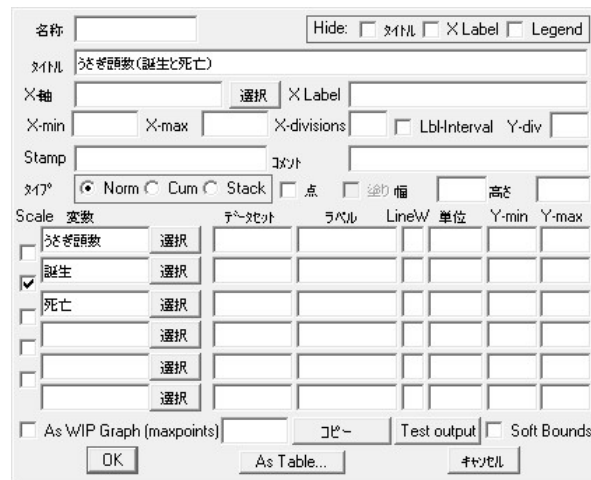



「うさぎ頭数」は、シミュレーション「減少」で減少します。なぜなら、「平均寿命」が短く設定されているからです。これによって、「誕生」に関連する正のフィードバックループより「死亡」に関連する負のフィードバックループが強力になり、その結果、「うさぎ頭数」が時間とともに減少します。


#### 6.7.7 入出力オブジェクト

入出力オブジェクトは、スケッチにスライダやグラフ、表などをはめ込む時に使います。実際に、作成したグラフをはめ込んでみます。

- 制御パネル  をクリックし、[ グラフ ] タブを選びます。
  - [ 新規 ] を選ぶと、グラフ編集ダイアログが開きます。
  - タイトルを「うさぎ頭数 (誕生と死亡)」と入力します。
  - 最初の [ 選択ボタン ] から、「うさぎ頭数」を選択します。
  - 次の [ 選択ボタン ] から、「誕生」を選択します。
  - 3 番目の [ 選択ボタン ] から、「死亡」を選択します。
  - 「誕生」と「死亡」の間にある [ Scale【目盛り】 ] のチェックボックスにチェックを入れます。
- 編集ダイアログは、以下ようになります。




- [ OK ] をクリックします。
- 入出力オブジェクト  をクリックします。

注意 もし統合シミュレーションを実行したままだと、スケッチツールバーが薄く表示されています。その場合は、停止  をクリックしてから、もう一度やり直してください。

- スケッチにおいて、図の空白部分をクリックします。  
入出力オブジェクトのダイアログが開きます。



グラフのタイトルに基づき、グラフの名前が作成されています。グラフの名前に文字数の制限があるので始めの部分が使われています。

- [ 自作グラフ出力 ] のラジオボタンをクリックします。
- [ 自作グラフ出力/分析ツール出力選択 ] のドロップダウンボックス [ ] をクリックし、「うさぎ頭数(誕生)」を選択します。
- [ OK ] をクリックします。
- グラフの位置とサイズを調節します。
- シミュレーション名をダブルクリックし、「実験」と入力します。
- 統合シミュレーション  をクリックします。
- スライダを動かすことで、グラフも変化します。

12 章では、入力と出力について、さらに詳しく説明します。

## 第7章

# 関数とシミュレーションエラー

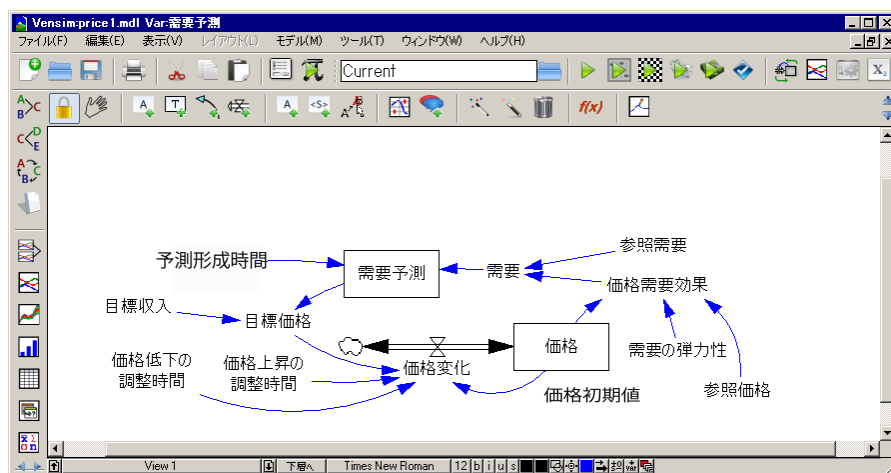
前章で扱ったうさぎ頭数モデルは、掛け算、割り算だけを使った簡単なモデルです。方程式では、加減乗除が一般的に用いられます。しかしそれだけでない関係が必要になることもあります。Vensim は、方程式で利用するさまざまな関数を持っています。本章では、これらの関数を方程式に用いる方法を紹介します。さらに、曲線グラフを用いた関数的な関係を取り扱う表関数を紹介します。

さらに“浮動小数点エラー”という問題への対応について見ていきます。この問題は、数値が大きすぎた時やゼロで割られた時、または関数において想定範囲を超えた値を受け取った時などに発生します。

### 7.1 価格と需要



例として、価格と需要に関する簡単なモデルの一部を使います。このモデルは、完成したものではありません。価格は、一定の収益を得ることを条件とした設定が行われています。本来、需要は価格に左右されるものですが、このモデルでは、需要は価格から影響されないという前提で作成されています。このモデルはダイナミックな仮説を示したものです。その結果が現実的でなくても、個々の実験をする上では興味深いものです。

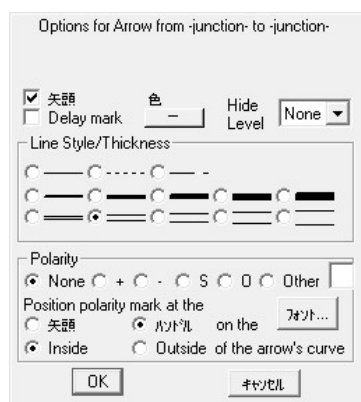
➤ 以下の図を作成するか、`\guide\chap07\complete\price1.mdl` を開きます。



サンプルモデルは英語です。英語の変数名を変数 **A** または、ストック変数 **T** をつけて図のような日本語表記に変更することができます。


このモデルを作成するには、6章で学んだ作図方法を使います。しかし、2カ所違う図があります。1つは、双方向のフロー「価格変化」です。2つは、ストック「需要予測」へのフローがないことです。双方向のフローを描くには次のようにします。

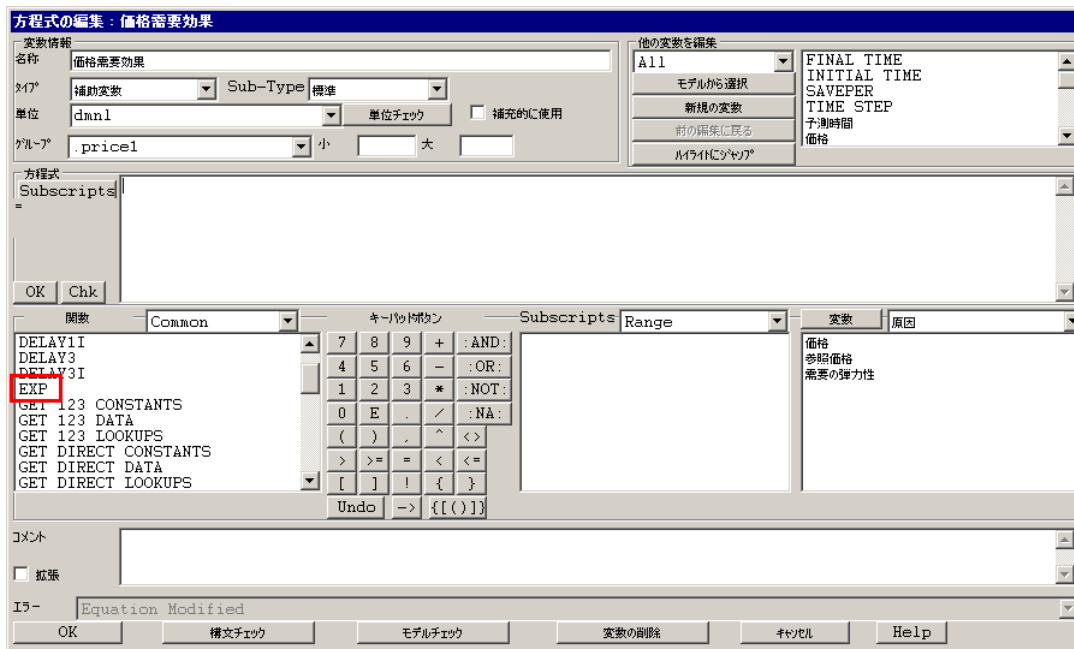
- フロー  を使って、「価格」の左から「価格」に向かうフローを作成し、「価格変化」と名付けます。
- スケッチ移動  をクリックします。
- 雲と「価格変化」を結ぶパイプの中央にあるハンドルを、右クリックするか、**[Ctrl]**を押しながら左クリックします。
- **[矢頭]** にチェックを入れます。




他の矢印の作成方法と同じように、「需要予測」に向かう矢印を作成します。

## 7.2 関数の入力






- 方程式  を選びます。
- 「価格需要効果」をクリックします。
- 左側の **[関数]** タブをクリックします。



利用可能な関数のリストが表示されます。リストの中のいずれかをクリックします。また関数名の最初の一文字を入力すると、その文字で始まる最初の関数に移動することができます。上下の矢印キーを使って、関数のリスト間を移動することもできます。いずれの方法でも、選択された関数は、方程式編集ダイアログの下にある [エラー] に、その関数の構文が表示されます。

- 関数のリストの中から、[ EXP ] をクリックします。
-  を押します。

編集ボックスの中には、{x} が強調表示された、「EXP({})」という文字が入力されます。

-  を押すか、編集ダイアログの中央にあるキーパッドの [ - ] をクリックします。
- [ 変数 ] タブをクリックし、「需要の弾力性」をクリックします。
-  を押すか、編集ダイアログのキーパッドから [ \* ] をクリックします。
- [ 関数 ] タブをクリックします。
- 関数のリストから、[ LN ] をクリックします。  を押すと、その関数を素早く探せます。  を押します。
- [ 変数 ] タブをクリックします。
- 「価格」をクリックします。
-  を押すか、編集ダイアログのキーパッドから [ / ] をクリックします。
- 「参照価格」をクリックします。

以下のような方程式となります。

$$\text{価格需要効果} = \text{EXP}(-\text{需要の弾力性} * \text{LN}(\text{価格} / \text{参照価格}))$$

関数のリストを使う代わりに、編集ボックスに直接キーボード入力することもできます。この方法が容易な場合もあります。

関数については、[ ヘルプ ] メニュー [ Vensim マニュアル ] で、ヘルプを起動し、[ キーワード ] タブで「exp」とすると、その構文が表示されます。ただし英語です。

この方程式の中には、2つの関数があります。「LN」は、数値の自然対数を返します。自然対数とは、定数  $e$  (2.72) を底とする対数のことです。「EXP(x)」は  $e$  の  $x$  乗を返します。

この方程式は、以下のようにも表せます。

$$\text{価格需要効果} = \text{POWER}(\text{価格}/\text{参照価格}, -\text{需要の弾力性})$$

$$\text{価格需要効果} = (\text{価格}/\text{参照価格})^{(-\text{需要の弾力性})}$$

この方程式は、一般的な一定の弾力性を持つ需要曲線です。「価格」/「参照価格」は価格の標準化を行っており、単位エラーの発生を防ぐ役割もあります。標準化については、次の章でさらに詳しく述べます。

どの関数を使うにあたって、その入力方法は同じです。関数を直接タイプするか、もしくはリストの中から選びます。

#### Excel による数式の確認

Vensim の関数は、Excel の関数と共通点があります。Excel には、LN、EXP、POWER という関数があり、Vensim と同じ構文です。

「参照価格」を 100、「需要の弾力性」を 1 とし、「価格」が 80 から 120 まで変化するとします。Excel 関数を使って、先に示した Vensim の方程式を表現することができます。

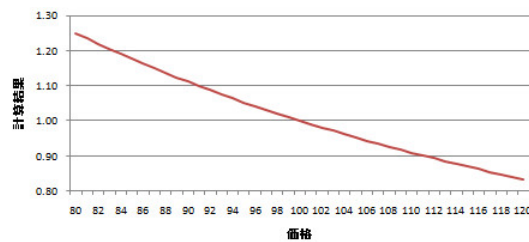
	A	B	C	D	E	F	G	H	I
1	価格	参照価格	需要の弾力性	LN(価格/参照価格)	-需要の弾力性*D	数式1	数式2	数式3	
2	A	B	C	D	E	Exp(E)	Power(A/B,-C)	(A/B)^-C	
3	80	100	1	-0.22	0.22	1.25	1.25	1.25	
4	81	100	1	-0.21	0.21	1.23	1.23	1.23	
5	82	100	1	-0.20	0.20	1.22	1.22	1.22	
6	83	100	1	-0.19	0.19	1.20	1.20	1.20	
7	84	100	1	-0.17	0.17	1.19	1.19	1.19	
8	85	100	1	-0.16	0.16	1.18	1.18	1.18	
9	86	100	1	-0.15	0.15	1.16	1.16	1.16	
10	87	100	1	-0.14	0.14	1.15	1.15	1.15	
11	88	100	1	-0.13	0.13	1.14	1.14	1.14	
12	89	100	1	-0.12	0.12	1.12	1.12	1.12	
13	90	100	1	-0.11	0.11	1.11	1.11	1.11	
14	91	100	1	-0.09	0.09	1.10	1.10	1.10	
15	92	100	1	-0.08	0.08	1.09	1.09	1.09	

Excel を数式表示したものが次の図であり、関数の利用方法がわかります。

	A	B	C	D	E	F	G	H
1	価格	参照価格	需要の弾力性	LN(価格/参照価格)	-需要の弾力性*D	数式1	数式2	数式3
2	A	B	C	D	E	Exp(E)	Power(A/B,-C)	(A/B)^-C
3	80	100	1	=LN(A3/B3)	=-C3*D3	=EXP(E3)	=POWER(A3/B3,-C3)	=(A3/B3)^-C3
4	81	100	1	=LN(A4/B4)	=-C4*D4	=EXP(E4)	=POWER(A4/B4,-C4)	=(A4/B4)^-C4
5	82	100	1	=LN(A5/B5)	=-C5*D5	=EXP(E5)	=POWER(A5/B5,-C5)	=(A5/B5)^-C5
6	83	100	1	=LN(A6/B6)	=-C6*D6	=EXP(E6)	=POWER(A6/B6,-C6)	=(A6/B6)^-C6
7	84	100	1	=LN(A7/B7)	=-C7*D7	=EXP(E7)	=POWER(A7/B7,-C7)	=(A7/B7)^-C7
8	85	100	1	=LN(A8/B8)	=-C8*D8	=EXP(E8)	=POWER(A8/B8,-C8)	=(A8/B8)^-C8
9	86	100	1	=LN(A9/B9)	=-C9*D9	=EXP(E9)	=POWER(A9/B9,-C9)	=(A9/B9)^-C9
10	87	100	1	=LN(A10/B10)	=-C10*D10	=EXP(E10)	=POWER(A10/B10,-C10)	=(A10/B10)^-C10

また「価格」に対応した計算結果をグラフにすると次の図のようになります。このグラフは、「価格」が「参照価格」よりも安価であれば需要が高まり、反対に割高感があれば需要が減少するという関係を示しています。





### 7.2.1 IF THEN ELSE 関数

状況によって、適用する条件を変更したいということはよくあります。IF THEN ELSE 関数は、これを可能にします。

a の条件が満たされれば b が返り、a の条件が満たされない場合は c が返るという式は、通常次のようになります。

$d = \text{IF } a \text{ THEN } b \text{ ELSE } c$

Vensim では、これが以下のような構文になります。

$d = \text{IF THEN ELSE (a,b,c)}$

- 方程式 **f(x)** を選び、「価格変化」をクリックします。
- [関数] タブをクリックし、リストから [IF THEN ELSE] を選び、**[↓]**を押します。
- [変数] タブをクリックします。
- 「目標価格」をクリックし、**[>]**を押します。(または、[(不) 等号等] タブをクリックし、[>] をクリックします) 次に、「価格」をクリックします。
- 方程式の中の{ontrue}をダブルクリックし、青い強調表示にします。
- **[Ctrl]+[↓]**を押します。
- **[ ]**をおして、「目標価格」をクリックし、**[=]**を押します。「価格」をクリックし、**[ ]**を押します。**[ / ]**を押し、「価格上昇の調整時間」をクリックします。
- 方程式の中の{onfalse}をダブルクリックし、青い強調表示にします。
- **[Ctrl]+[↓]**を押します。
- **[ ]**をおして、「目標価格」をクリックし、**[=]**を押します。「価格」をクリックし、**[ ]**を押します。**[ / ]**を押し、「価格低下の調整時間」をクリックし **[ ]**を押します。

方程式は、以下のようになります。

価格変化 = IF THEN ELSE(目標価格>価格,  
(目標価格-価格)/価格上昇の調整時間,  
(目標価格-価格)/価格低下の調整時間)

この関数の構文はこのように入力します。その引数を、ダブルクリック後に入力することで、引数として入力されます。引数は中括弧{}で囲まれています。それはコメントとして扱われますが、引数を入力した後、削除した方が良いです。**[Ctrl]+[↓]**を使うと、方程式に改行を挿入でき、式が見やすくなります。

### 7.2.2 SMOOTH 関数

SMOOTH 関数は、平均的に必要な時間や予測を表現するために使われる関数です。SMOOTH が、LN や EXP および IF THEN ELSE と異なるのは、時間の変化を組み込んでいるという点です。つまり、もしある  $x$  の値があれば、 $\text{EXP}(x)$  を計算することができます。しかし、 $x$  の値だけでは  $\text{SMOOTH}(x,4)$  の値はわかりません。つまり SMOOTH が事前にどのような値を持っていたのかを知る必要があります。なぜなら SMOOTH 関数は、ストックに近い働きを持つ関数であるともいえるからです。


次のような方程式を作成します。

需要予測 = SMOOTH(需要, 予測時間)

この方程式は以下とまったく同じものです。

需要予測 = INTEG((需要-需要予測)/予測時間, 需要)


smooth 方程式を入力します。

- 「需要予測」の方程式編集ダイアログを開きます。
- 「需要予測」は、ストックとして作成されているため、編集ダイアログを開いた時、[タイプ] が [ストック] として表示されます。SMOOTH の方程式を入力するために、[補助変数] に変えます。しかしながら、ストックとしての意味も持つため、箱で囲まれた変数のままにします。
- [タイプ] を、左側のドロップダウンボックス [ ] で、[ストック] から [補助変数] に変えます。
- [関数] タブをクリックし、[SMOOTH] を選んで  を押します。
- [変数] タブをクリックし、「需要」をクリックします。
- 方程式の中の {stime} をダブルクリックし、変数のリストから「予測時間」をクリックします。
- [単位] に「箱/月」を加えます。「需要」と同じ単位です。
- [OK] をクリックし、編集ダイアログを閉じます。

SMOOTH に似た関数があります。例えば、DELAY1、DELAY1I、DELAY3、DELAY3I、FORECAST、SMOOTH3、SMOOTH3I、SMOOTHI および TREND などです。隠れたストック的な意味を強調するために箱に囲まれた変数として表示することは有効です。SMOOTH 関数は単純なものです。この SMOOTH の代わりに積分形式による表現 (INTEG) が理解しやすい場合もありますが、これは好みの問題です。

SMOOTH 関数  $(x,t)$  の形式であっても、積分形式の  $(\text{INTEG}(x-sx)/t,x)$  として書かれていても、図では、フローではなく情報の矢印となります。ストックがフローなしで用いられることを認めるか、認めないかは賛否があります。Vensim では、「需要予測」のような情報の概念を表現するためにこうした表現をすることがあります。この場合、情報を増加させたり減少させたりする物理的なプロセスはありません。それは、圧力やアンバランスに対する調整です。Vensim ではストックに入るフローだけでなく、情報の矢印も描くことができます。したがって、表現方法のルールは、ユーザーが決める必要があります。

### 7.2.3 STEP 関数

- 「目標収入」の編集ダイアログを開きます。
- 「10000+」と入力します。
- [ 関数 ] タブをクリックします。
- 関数のリストから、[ STEP ] を選んで  を押します。
- 「5000」と入力します。
- 方程式の中の{stime}をダブルクリックします。
- 「10」と入力します。
- [ 単位 ] に「円/月」と入力し、[ OK ] を押します。

方程式は、以下のようになります。

$$\text{目標収入} = 10000 + \text{STEP}(5000, 10)$$

STEP は、入力関数として使われます。この関数は、時間が{stime}に達するまでゼロを返し、それから後は{sheight}を返します。この例では、時間が 10 になるまでは、ゼロを返し、それからは 5000 を返します。そのため、「目標収入」は 10,000 円から始まります。時間が 10 まではその値は一定です。その後は 15,000 円にはね上がります。STEP 関数は、モデルが作り出す振る舞いを見る上で、とても良い方法です。例えばサプライチェーンのモデルにおいて、需要の急激な変化が供給に及ぼす影響を調べるなどに使われます。

STEP 関数に似た関数があります。PULSE、PULSE TRAIN、RAMP および RANDOM です。これらの関数はすべて、異なる時間に、別の値を返します。RANDOM 関数は、時間ごとに異なる値を返します。しかし実行された結果は同じであるため、結果を繰り返すことができます。

もう一つの時間的な変化をもたらすものに Data があります。これは 16 章の「モデルにおけるデータの利用」で紹介します。異なるタイプの振る舞いを得るために、「Time」という変数を SIN 関数の引数とすることもできます。

### 7.2.4 初期条件

「価格」の初期値は、「初期価格」によって与えられます。しかし、「初期価格」から「価格」への矢印はありません。「価格」の編集ダイアログを開いた時、「初期価格」は変数のリストに表示されていません。

- [ 変数をすべて選択 ] をクリックします。( Vensim PLE と PEL Plus は対応しません )
- リストから「初期価格」を選び、[ OK ] をクリックします。
- 単位に「円/箱」と入力します。
- [ OK ] をクリックして、編集ダイアログを閉じます。

注意 「価格」から「初期価格」への矢印を作成していた場合、「初期価格」が変数リストに表示されています。しかし、編集ダイアログを閉じた時に、矢印が消えます。なぜならデフォルトでは、初期原因は、モデル図に表示されないからです。この矢印を残したい場合、[モデル]メニュー [設定]を選び、[スケッチ]タブの中の、[Show initial causes on model diagrams]【モデル図における初期原因を表示する】にチェックを入れてください。

以下は、このモデルの完成した方程式です。ほとんどが、簡単なものです。  
`uide\chap07\complete\price2.mdl`にあるモデルがこれです。

需要 = 参照需要\*价格需要效果  
Units : 箱/月

```
需要の弾力性 = 1
Units : dmnl

価格需要効果 = EXP(-需要の弾力性*LN(価格/参照価格))
Units: dmnl

需要予測 = SMOOTH(需要, 予測時間)
Units : 箱/月

FINAL TIME = 100
Units : 月

初期価格 = 100
Units : 円/箱

INITIAL TIME = 0
Units : 月

価格 = INTEG (価格変化, 初期価格)
Units : 円/箱

参照需要 = 100
Units : 箱/月

参照価格 = 100
Units : 円/箱

SAVEPER = TIME STEP
Units : 月

目標価格 = 目標収入/需要予測
Units : 円/箱

目標収入 = 10000+STEP(5000,10)
Units : 円/月

TIME STEP = 1
Units : 月

価格低下の調整時間 = 6
Units : 月

価格上昇の調整時間 = 4
Units : 月

予測時間 = 6
Units : 月
```

## 7.4 シミュレーションエラー



作成したモデルをシミュレーションする準備ができました。

- [モデル]メニュー [モデルチェック]を選びます。または、**Ctrl**+**T**を押します。[モデル OK です]というメッセージが表示されます。もしこのメッセージが表示されない場合、問題のある方程式の編集ダイアログが開きます。シミュレーションを実行する前に、間違いを修正します。
- [モデル]メニュー [単位チェック]を選びます。または、**Ctrl**+**U**を押します。

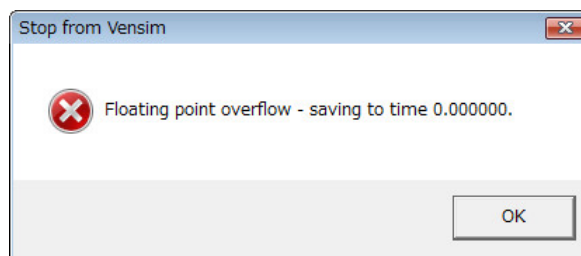
[単位 OK です] というメッセージが表示されます。もしこのメッセージが表示されない場合、間違いを修正します。

次は、このモデルで故意にシミュレーションエラーを発生させます。そして、その解決方法を見ていきます。

### 7.4.1 時間がゼロのエラー

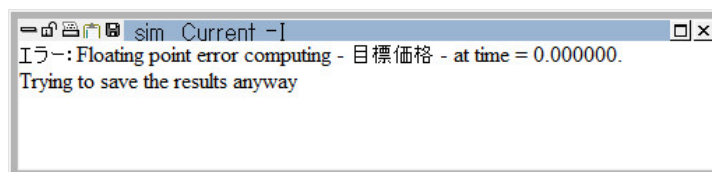
- シミュレーションの準備  をクリックします。
- 「参照需要」をクリックし、値を「0」と入力します。
- シミュレーションの実行  をクリックします。

エラーメッセージが表示されます。←上書きしますか「O.K」



- [OK] をクリックします。





次のようなエラーウィンドウが現れます。




このウィンドウは、問題が見つかった時、どの変数が計算されていたかを示しています。エラーは時間がゼロで発生しています。そこで表ツールを使って、問題を確認します。


- 「目標価格」をクリックし、ワークベンチ変数にして下さい。

Vensim PLE もしくは PLE Plus の場合は、以下の方法をとります。

- 表  をクリックします。
- 因果ツリー  をクリックします。
- 因果ツリーの出力結果にある「需要予測」をクリックします。
- 表  をクリックします。
- 因果ツリーの出力結果にある「目標収入」をクリックします。
- 表  をクリックします。

Vensim Standard、Runtime、Professional または DSS を使っている場合は、

- 表  を右クリックします。

- [ Show Link or Sketch Vars ] のドロップダウンボックス [ ] から [ Causes ] 【原因】を選びます。
- 表  をクリックします。

以下のような表示が現れます。

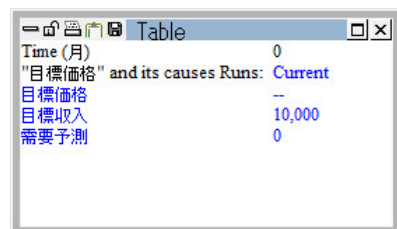


Table	
Time (月)	0
"目標価格" and its causes Runs:	Current
目標価格	--
目標収入	10,000
需要予測	0





「目標価格」のとなりに「--」という値があります。これは「目標価格」が計算されていないことを示しています。「需要予測」と「目標収入」は計算されており、それらの値が表示されています。「目標価格」の方程式は以下のようになっています。

$$\text{目標価格} = \text{目標収入} / \text{需要予測}$$

ゼロで割ったためにエラーが発生したことがわかります。「需要予測」がゼロになる理由を探ると、「参照需要」がゼロのために、「需要」がゼロになったことがわかります。

たとえ「需要予測」がゼロであっても「目標価格」にエラーを生じさせない方法があります。それは本章最後の「モデルの修正」で紹介します。

#### 7.4.2 シミュレーション中のエラー

- シミュレーションの準備  をクリックします。
- 「需要の弾力性」をクリックし、値を「2」と入力し、 を押します。
- 「予測時間」をクリックし、値を「1」と入力し、 を押します。
- シミュレーションの実行  をクリックし、[ はい ] をクリックして、上書きします。

先程表示されたのと同じような、以下のメッセージが表示されます。


```
ERROR: Floating point error computing - 目標価格 - at time = 30.000000.
Trying to save the results anyway
```


先程と同じ分析方法にあてはめて、この問題を解決できます。以下は、「目標価格」とその原因の表です。

Time (月)	29	30
"目標価格" and its causes Runs:	Current	
目標価格	2.26674e+027	2.26674e+027
目標収入	15,000	15,000
需要予測	6.61744e-024	0

(20月と30月はスクロールバーを操作して表示させてください)




ここに表示された表から、2つのことがわかります。1つは、インプットされた数値が異なっても、29月と30月で「目標価格」が、同じ値を示しているということです。シミュレーションの実行中にエラーが発生した場合、完全には計算されていない変数の値は、その時点より前のものになります。2つは、エラーの原因は先程のケースと同じということです。

シミュレーションの実行中にエラーが発生した後、何が問題を引き起こしたか、直接原因グラフ  を使って知ることができます。


- 「目標価格」をクリックし、ワークベンチ変数とします。
- 直接原因グラフ  をクリックします。
- 「需要予測」をクリックし、その原因をさぐります。次に、「需要」、「価格需要効果」、「価格」、「価格変化」の原因をグラフ表示します。最後に「価格変化」は「目標価格」に戻ってきます。

このように「価格」の上昇による「需要」の減少、「需要予測」の減少、目標収入を確保するための「目標価格」の情報という正のフィードバックがあります。これは指数関数的な増加であり、急速に発生するために、エラーが生じます。

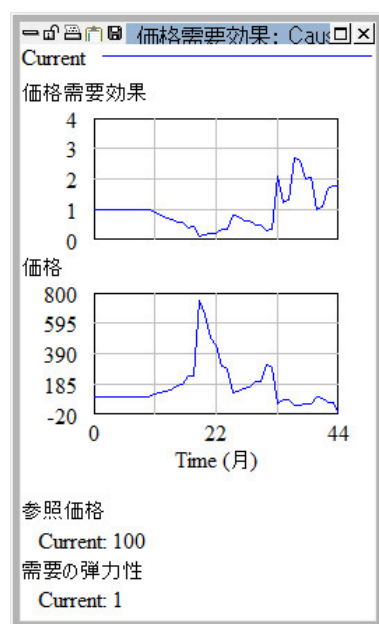
### 7.4.3 不可解な変化

- シミュレーションの準備  をクリックします。
- 「予測時間」をクリックし、値を「0.5」と入力し、を押します。
- シミュレーションの実行  をクリックし、[はい] をクリックして、上書きします。


「価格需要効果」の時間が44のところで、エラーが発生していることがわかります。

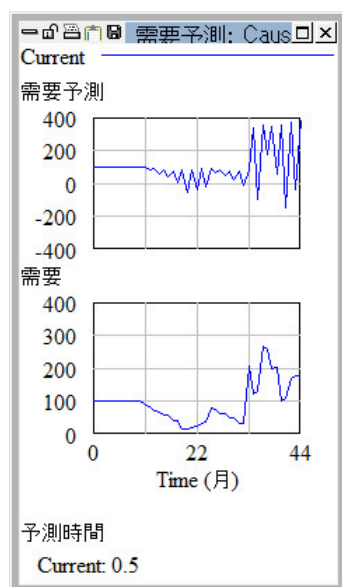
- 「価格需要効果」をクリックし、直接原因グラフ  をクリックします。





値は大きく上下に変化していますが、エラーの原因は、最後に「価格」がたどり着く負の値です。

- 直接原因グラフ  を使って振る舞いの原因を調べます。「価格需要効果」、「価格」、「価格変化」、「目標価格」を経て「需要予測」にたどり着きます。



「需要」が常に正の値を示していても「需要予測」は負の値を示します。これは「需要予測」に対する時定数（つまり予測時間）が、[時間ステップ]よりも短かったためです。

このような奇妙な結果は、こうしたことに原因する場合があります。その場合は、[時間ステップ]をより小さい値とします。その結果がもし変更前と大きく違うようであれば、時定数が小さすぎるのが原因です。

需要は、予測時間だけの遅れをとまって、需要予測に反映されます。したがって、需要と需要予測は似たグラフとなるべきです。

[モデル]メニュー [設定]で確認できます。このモデルでは“1月”です。

#### 7.4.4 エラーの統合

以下のような、もう 1 つのエラーを見てみます。




```
ERROR:
Floating point error integrating - Big Level - at time = 340.0
Trying to save the results anyway
```

このエラーは、すべての方程式が計算されていたとしても、フローの値が非常に大きいことを示しています。このようなエラーは、正のフィードバックループがあるときに、シミュレーション中のエラーとして生じます。

### 7.5 モデルの修正

このモデルをより頑健なものにするための修正方法があります。「需要予測」がゼロでも、正確な振る舞いを示す「目標価格」の方程式を設定します。この修正されたモデルは、`\guide\chap07\complete\price3.mdl` にあります。

#### 7.5.1 XIDZ

- 変数  をクリックし、「目標価格」の左の空白に、「最大価格」を作成します。
- 矢印  をクリックし、「最大価格」から「目標価格」への矢印を描きます。
- 方程式  を選択し、「目標価格」をクリックします。以下のような、方程式を入力します。

```
目標価格 = MIN(XIDZ(目標収入, 需要予測, 最大価格), 最大価格)
Units : 円/箱
```

- 「最大価格」に方程式を入力します。

```
最大価格 = 25000
Units : 円/箱
```

XIDZ 機能は、ゼロ以外の値で割られた商であり、ゼロで割る場合は第 3 引数を戻します。つまり、以下の 2 つは、「需要予測」がゼロである場合を除いて、同じものになります。

```
XIDZ(目標収入, 需要予測, 最大価格)
```

```
目標収入/需要予測
```

「需要予測」がゼロの場合、XIDZ を用いた場合は「最大価格」が戻され、単なる割り算による式ではエラーが生じます。

XIDZ 関数は、変数がゼロという値をとる可能性があり、それでもシミュレーションを続けたい場合には有効なものです。

また関係の深い機能として、ゼロで割られたときゼロを返す ZIDZ 関数があります。

## 第 8 章

# 表関数の作成

7 章では、いくつかの関数を使ったモデルを作成しました。さまざまな関係を表すためには、組み込み関数だけではなく、必要に応じて、関数を作成した方がよい場合もあります。

表関数は、2 つの変数間のカスタマイズされた関係を定義します。方程式としては、以下のように定義されます。

$$y = \text{my function}(x)$$

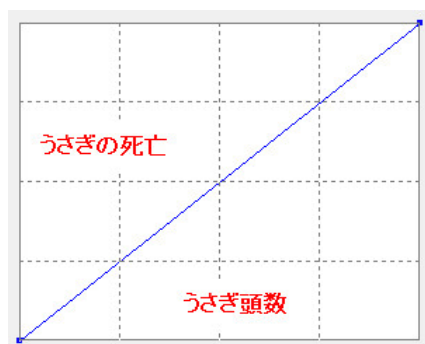
出力される変数  $y$  は、入力する変数  $x$  が、あなたが指定した形（非線形の場合が多い）を持つ表関数 `my function` を通じて変化したものです。

表関数は、グラフ関数と言われることもあります。これらは、数字の表（方程式編集ダイアログの中）、もしくはグラフ（表関数グラフ編集ダイアログの中）で作成されます。

### 8.1 うさぎ増加の限界

ここでは、6 章のうさぎ頭数モデルを、養える頭数に限界がある環境を想定したものへと拡張します。6 章で示した制限のない頭数増加モデル（`pop.mdl`）は、環境に応じた頭数限界を持つモデル（`rabbit.mdl`）になります。

`pop.mdl` の「死亡」の値は、うさぎの頭数に正比例します。実際に、「うさぎ頭数」と「死亡」の関係は、線形の関係です。

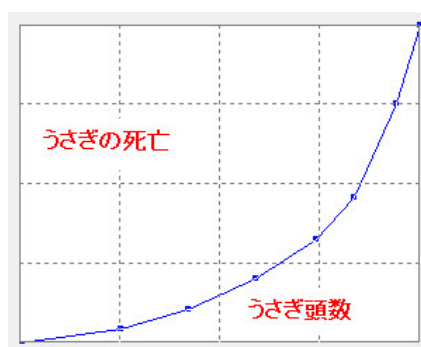


これは、「死亡」が時間と共に線形的に増加することを意味していません。線形とは、「死亡」が「うさぎ頭数」と同じように増加することです（もし「うさぎ頭数」が指

数的に増加した場合、「死亡」も指数的に増加します)。

この線形の関係を持つ、上記のような表関数を作成します。しかし、「うさぎ頭数」を掛け合わせる定数（表関数の傾きと同じ値を持つ定数）を使用するほうが簡単です。実際、6章のモデルは、「うさぎ頭数」/「平均寿命」を使用しており、その結果、直線の傾きは、 $1/\text{「平均寿命」}$ となります。この方法は、「平均寿命」を理解する方が、曲線の形状を理解するよりも易しいためです。

このモデルで示したいことは、「うさぎ頭数」が多くなるにつれて、「死亡」が「うさぎ頭数」よりも早く増えるということです。これは、「うさぎ頭数」が増加すれば、資源（食物など）の限界に近づくため、平均すれば早く死亡するためです。つまり、以下のような関数が必要になります。



表関数を使ってこのような関数を作成します。その前に、表関数の入出力を標準化する必要があります。

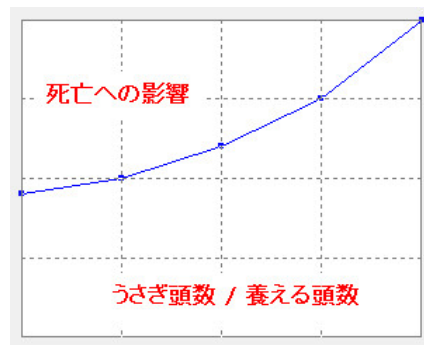
## 8.2 表関数の標準化

上記のグラフ関数は、入力値としての「うさぎ頭数」を、出力値としての「うさぎの死亡」を表しています。このグラフは、作成が難しいだけでなく、修正も難しいものです。例えば、長生きする新種が導入された時の変化を見るには、すべての関数をやり直す必要があります。もしくは、養える頭数が増加した場合の効果を見るにも、すべての関数をやり直す必要があります。

標準化された入出力の値は、ゼロから1の間をとります。入力値は、次元がないだけでなく、サイズや単位からも独立します。例えば、千頭単位の「うさぎ頭数」から千頭単位のその1ヶ月での「死亡」を計測するとします。「うさぎ頭数」を入力値とし、「死亡」を出力値とした表関数は、妥当なものではありません。

一方で、入力値である「うさぎ頭数」を「養えるうさぎ頭数」に関連させて標準化した表関数では、「死亡への影響」という指標、または死亡数（「うさぎ頭数」/「平均寿命」）といった出力値に対して、サイズや単位が変更になっても修正する必要がありません。

標準化によって、一般化された表関数を作成することができます。もし、「うさぎ頭数」のサイズや性質が変化した場合、表関数そのものを修正するのではなく、「養えるうさぎ頭数」か「平均寿命」の値を変更します。





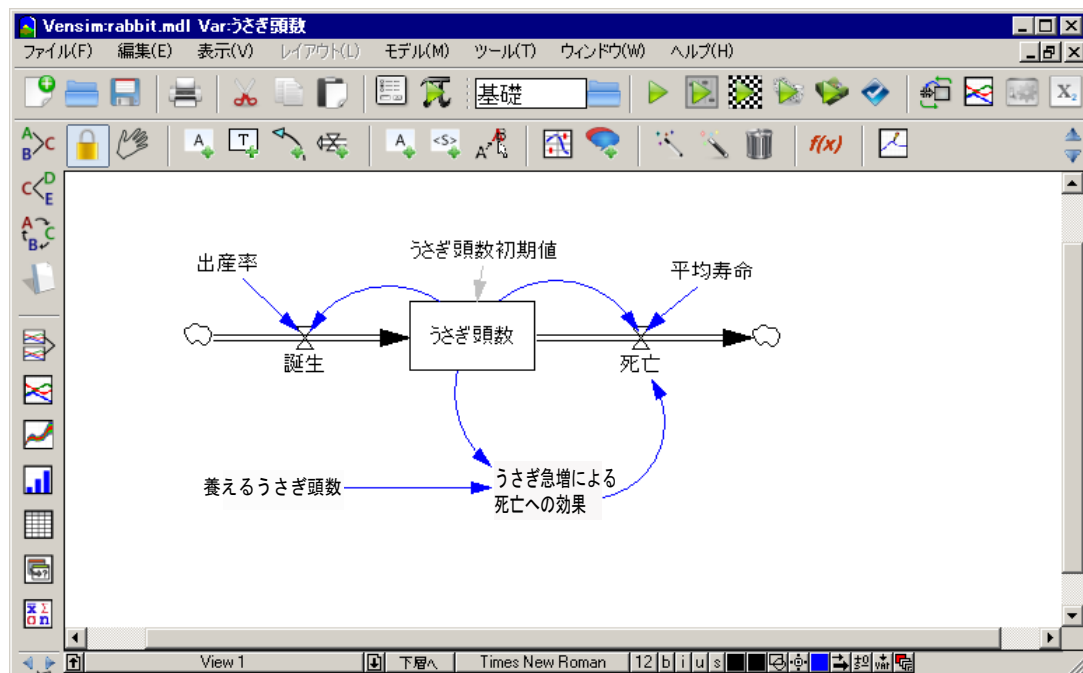
標準化するためには、入力した変数を標準値または平均値で割ります（例えば「うさぎ頭数」/「養えるうさぎ頭数」）。実際の「うさぎ頭数」が標準値と等しい時、入力値（参照元となる値）は1です。「うさぎ頭数」の値によっては、入力値は1より高かったり、低くなったりします。表関数からの出力値は、他の変数に影響します。別の表現をすると、入力値が標準値か平均値と等しければ、表関数からの出力値は1となり、入力値と出力値に関係がないことを意味します。

### 8.2.1 モデルの作成 (rabbit.mdl)


このモデル (*rabbit.mdl*) では、表関数を持つモデルを作成するための、簡単で、直接的な方法を紹介します。標準化は、表関数の出力変数「増えすぎによる死亡増」で行われます。そして、この変数の出力値は、「死亡」に直接影響します。標準化された入出力値を明示的に配置する例は、この章の後方部分にある「標準化された変数の分離」の節に詳しくあります。もしくは、`\guide\chap08\complete\rabbit2.mdl` を開きます。このモデルは、機能的には *rabbit.mdl* と同じですが、変数の数が増えています。分かりやすい方を選んでください。

英語版です。

- 新規モデル  をクリックします。
- モデル設定ダイアログの、[時間の範囲] タグの中にある [終了時間] に「30」と入力し、[時間ステップ] に「0.125」と入力するか、ドロップダウンボックス [▼] から [0.125] を選びます。次に [時間単位] のドロップダウンボックス [▼] から、[年] を選びます。
- [単位同義語] タブをクリックします。編集ボックスの中に、「頭, 頭数」と入力し、[編集して追加] をクリックし、[OK] をクリックします（もしくは  を押します）。
- 以下のようなモデルを作成します。
- モデル（例えば *rabbit.mdl*）を、`guide\chap08` に保存します。



### 8.2.2 方程式の入力

- 方程式  をクリックし、次に、以下の変数をそれぞれクリックし、方程式と単位を入力します。

うさぎ頭数 = INTEG(誕生 - 死亡, うさぎ頭数初期値)  
Units : 頭

うさぎ頭数初期値 = 1000  
Units : 頭

出産率 = 0.23  
Units : 1/年

平均寿命 = 8  
Units : 年

誕生 = うさぎ頭数 \* 出産率  
Units : 頭/年

死亡 = (うさぎ頭数 / 平均寿命) \* うさぎ急増による死亡への効果  
Units : 頭/年

養えるうさぎ頭数 = 1000  
Units : 頭

変数「養えるうさぎ頭数」は、この環境下で維持できるうさぎの最大値ではありません。むしろ、「養えるうさぎ頭数」は、その環境下でのうさぎの標準的な頭数を表しています。「養えるうさぎ頭数」の最大値を求めるには、別の公式が必要になります。

注意すべきことは、ストックの初期設定を数字を入力するのではなく、定数「うさぎ頭数初期値」を使って設定していることです。この方法だと、シミュレーションの実行中に値を変更することができます。また、「死亡」の方程式は「うさぎ急増による死亡への効果」と掛

け合わされています。

初期値として変数を追加するためには、方程式編集ダイアログの「変数」タブの、「変数をすべて選択」(Vensim PLE 及び PLE Plus は「Choose Initial Variable」【初期値の選択】)を使用します。

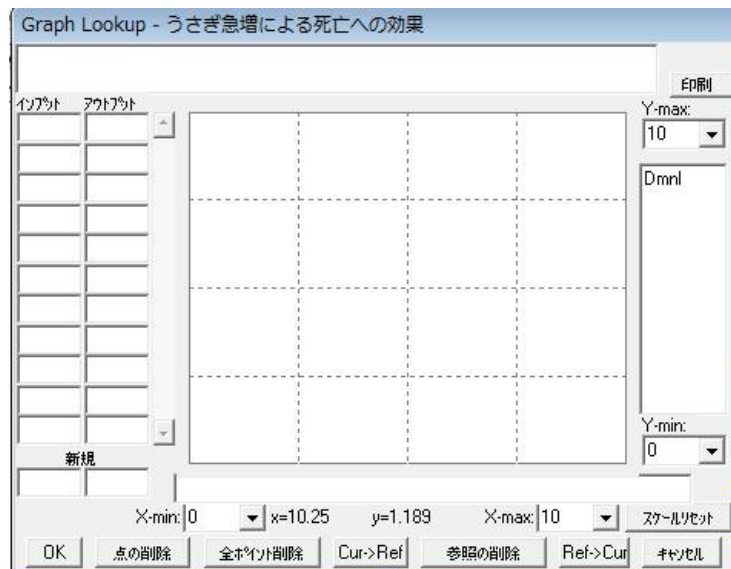
### 8.3 表関数の作成と標準化

- 方程式  $f(x)$  をクリックし、「うさぎ急増による死亡への効果」をクリックします。  
「タイプ」の下に、2つの「▼」を持つドロップダウンボックスがあります。1つは「補助変数」、もう1つは「標準」となっています。
- 「標準」のドロップダウンをクリックし、「表関数」に変えます。  
これで、この補助変数に、表関数が追加されます。
- 「変数」タブをクリックし、変数リストから「うさぎ頭数」をクリックします。
- $\frac{1}{x}$  を押し、「養えるうさぎ頭数」をクリックします。

増えすぎによる死亡増 = WITH LOOKUP(うさぎ頭数/養えるうさぎ頭数)

「うさぎ頭数」が変化することで、「うさぎ急増による死亡への効果」の値も表関数の形に従って変化します。表関数を表す、実際の表やグラフを作成します。

- 方程式編集ダイアログの「タイプ」の右横にある「As graph」をクリックします。  
表関数グラフ編集ダイアログが開きます。



- 画面左下にある「新規」の、左の欄に「0」を入力し、 $\leftarrow$ を押します。カーソルが右の欄に移動します。「0.9」と入力し、 $\leftarrow$ を押します。カーソルが左の欄に戻り、先ほど入力した「0」が、上の「インプット」「アウトプット」の列に新しい数値として入ります。
- 以下の値を続けて入力します。値を入力するごとに、 $\leftarrow$ を押します。グラフが自動的に描かれます。  
「0,0.9」、「1,1」、「2,1.2」、「3,1.5」、「4,2」

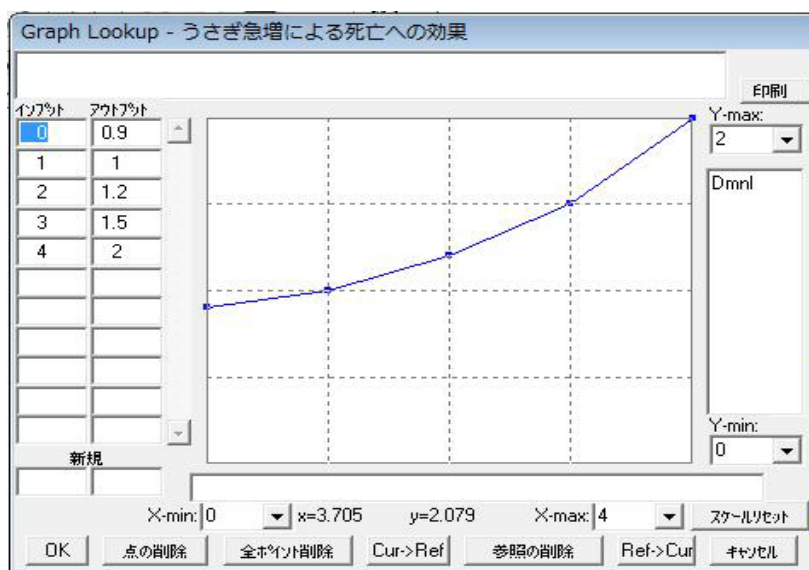
- もう1つの方法として、グラフ上でマウスをクリックして点を加え、グラフを描くこともできます。また、グラフ上で点をドラッグすることもできます。初めに、[X-max] と [Y-max] を、それぞれ「4」と「2」に設定しておきます。
- [スケールリセット] をクリックすることで、X と Y の目盛りを再設定します。

もしマウスで点を追加した場合、グラフに現れた点が、切りが良い数値になっていなくても心配いりません。曲線の形は、切りの良い値よりも重要です。しかし、「1,1」は正確に入力してください。なぜなら、「うさぎ頭数」と「養える頭数」とが等しい時に、参照される場所がこの点であるためです。そのとき、「死亡」への影響は生じません。

### 8.3.1 値の編集

- [インプット] [アウトプット] の値を、修正することができます。また、グラフ上で点をドラッグして修正することもできます。
- 点を消す場合、[点の削除] をクリックし、グラフ上で消去したい点をクリックします。最終的には、以下のようなグラフになります。

グラフ上では、カーソルが削除モードに変化します。




- [OK] をクリックし、表関数グラフ編集ダイアログを閉じます。

表形式で作成されたグラフ方程式が、[Lookup] の編集ボックスに、括弧つきで表示されます。これらの値は、表関数グラフ編集ダイアログで作成したのですが、直接入力することもできます。

- [単位] に「Dmnl」(Dimensionless のことで、単位が無いことを意味する。直接入力することも可能)を追加し、[OK] をクリックし、方程式編集ダイアログを閉じます。

Dimensionless は、大変重要な単位です。表関数における入力値の標準化では、「うさぎ頭数」を「養える頭数」で割りました。この2つの変数は両方とも単位が「頭」であり、単位を持たないことになります。



- 保存  をクリックして、モデルを保存します。

## 8.4 構文と単位のチェック

シミュレーションを実行する前に、方程式と単位のチェックをします。



- [モデル]メニュー [モデルチェック] を選びます。または **Ctrl**+**T** を押します。[モデル O.K. です] と表示されます。

間違いが見つかった場合、モデルの構造が、先に示した図と同じかどうか確認してください。構造が同じなら、変数に対して、方程式編集ダイアログを開いて、入力した方程式がリストと同じかどうか確認してください。

- [モデル]メニュー [単位チェック] を選びます。または **Ctrl**+**U** を押します。[単位 O.K. です] と表示されます。


エラーが表示された場合、どの変数に間違いがあるのかを確認します。問題のある変数の方程式編集ボックスを開いて、先ほどの方程式のリストから単位をチェックします。それでも単位に問題がある場合は、構文に間違いがあることも考えられます。

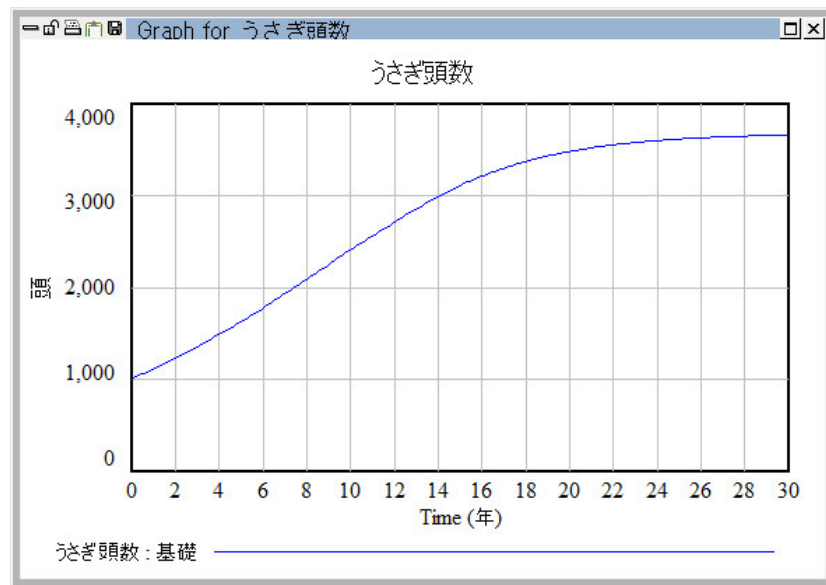
## 8.5 モデルのシミュレーション

- ツールバーにあるシミュレーション名の編集ボックス  をクリックし、最初のシミュレーション実行名を「基礎」と入力します。
- シミュレーションの実行  をクリックします。

シミュレーションの実行中は、進捗状況を示すウィンドウが表示されます。ただし、速度の速いコンピュータでは、このウィンドウは見えないかもしれません。

## 8.6 モデルの分析

- スケッチの「うさぎ頭数」をクリックし、「うさぎ頭数」をワークベンチ変数とします。タイトルバーに「うさぎ頭数」が表示されていることを確認してください。
- グラフ  をクリックします。「うさぎ頭数」のグラフが表示されます。

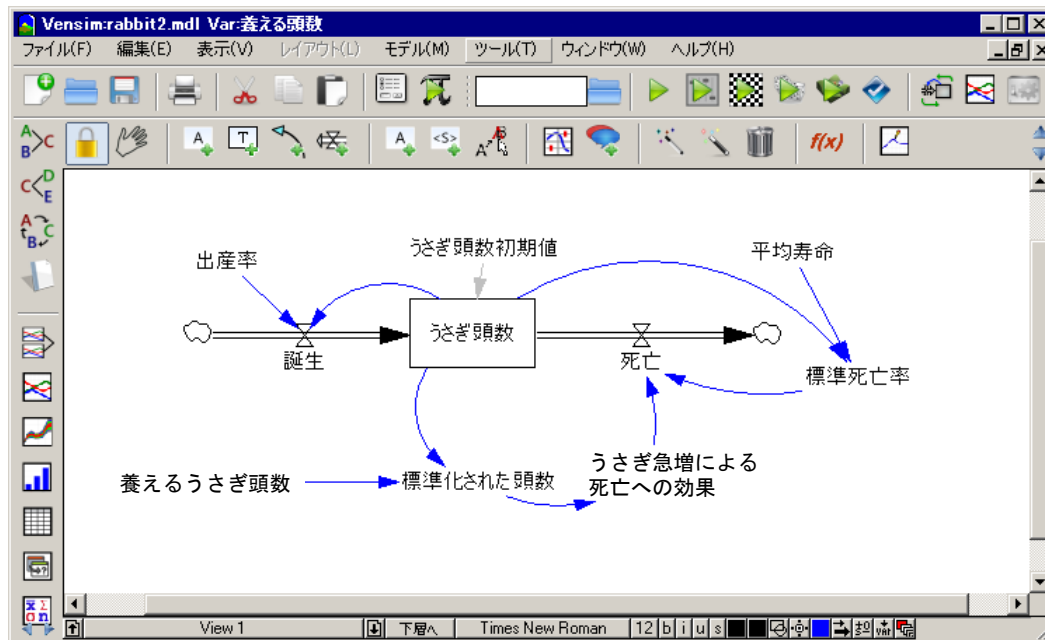


「うさぎ頭数」が、初めは指数的に増加し、その後緩やかな増加を伴って、約 3,500 頭の最大値に達していることがわかります。これは表関数による影響です。





## 8.7 標準化された変数の分離

この節はオプションです。すでに作成したモデル *rabbit.mdl* と同じシミュレーションを別の方法で作成します。


表関数は、変数が“標準値”より変化した場合に、モデルの変数を変化させる際に役立ちます。先ほどのモデル (*rabbit.mdl*) では、標準値は「死亡」のフロー方程式の中に含まれ、隠れていました。場合によっては、標準値を明示する必要があるかもしれません。標準値は定数のこともあります。以下の *rabbit2.mdl* モデルにおける「標準死亡率」のように補助変数のこともあります。



このモデルでは、「うさぎ急増による死亡への効果」の中で入力された変数を標準化するのではなく、明示的に分離した「標準化された頭数」という入力変数が与えられています。

- 先ほどのモデルを、別の名前 (*rabbit2.mdl*) で保存し、上のような図を作成します。
- 「平均寿命」から「死亡」に向かう矢印を、「標準死亡率」へと動かすために、スケッチ移動  を使って、「死亡」に向かう矢印の矢頭を「標準死亡率」へドラッグします。もしくは、消去  で矢頭をクリックして削除し、矢印  で新しい矢印を作成します。
- 方程式 *f(x)* をクリックし、「標準死亡率」をクリックします。以下の方程式（表関数から結果を出す前の「死亡」の方程式と同じです。）と単位を入力し、 を押します。

標準死亡率 = うさぎ頭数 / 平均寿命  
Units : 頭/年

- 「死亡」をクリックし、以下の方程式を入力し、 を押します。

死亡 = 標準死亡率 \* うさぎ急増による死亡への効果  
Units : 頭/年


- 「標準化された頭数」をクリックし、以下の方程式を入力します。

うさぎ急増による死亡への効果 = うさぎ頭数 / 養える頭数  
Units : dmm1

- 「うさぎ急増による死亡への効果」をクリックし、[WITH LOOKUP] の編集ボックスに、以下の方程式を入力します。[Look up] の編集ボックスと単位はそのままです。



うさぎ急増による死亡への効果 = WITH LOOKUP (標準化された頭数)  
Units : dmm1

方程式編集ダイアログを閉じると、変数の強調表示はなくなっています。

- ツールバーの保存  をクリックし、モデルを保存します。

注意すべきことは、標準化された「標準死亡率」は、この例のように補助変数だけでなく、定数にもなるという点です。



### 8.7.1 シミュレーション

- シミュレーションを実行する前に、単位チェック (**Ctrl**+**U**) と、モデルチェック (**Ctrl**+**T**) をします。
- シミュレーションの実行名を入力し、シミュレーションの実行  をクリックします。グラフ  で出力値を確認します。先ほどのモデルとまったく同じ振る舞いが得られます。


注目すべきことは、追加された変数の「標準死亡率」です。これは、頭数の密度が低く、「うさぎ急増による死亡へ効果」が影響しない場合の死亡数を計算します。先ほどのモデルでは、フローの「死亡」変数に、この計算を組み込みました。今回、わかりやすくするために別にし、“標準化された”変数によって表関数を用いる方法を紹介しました。

### 8.7.2 表関数の変更

一時的に表関数を変更し、モデルのシミュレーションを行います。

- シミュレーションの準備  をクリックします。
- シミュレーション名の編集ボックス  をクリックし、「実行2」と入力します。
- スケッチ上で黄色/青で表示された「増えすぎによる死亡増」をクリックします。

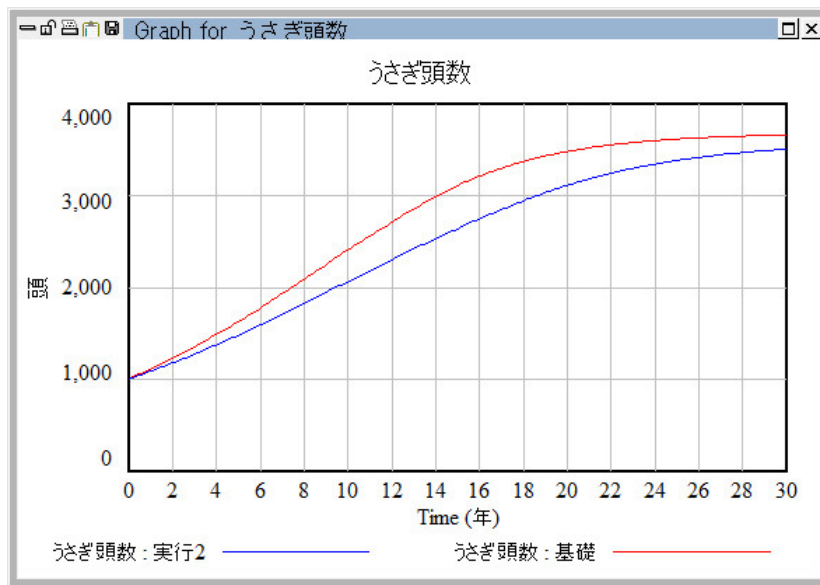
表関数グラフ編集ダイアログが表示されます。

- マウスでグラフ上の点を移動し、曲線の勾配を変更します。例えば、点を少し上に動かしてみます。目盛りの変更が必要な場合は、[Y-max:] [Y-min:] [X-max:] [X-min:] のドロップダウンボックス  をクリックして数字を選ぶか、直接入力します。点を増やしたり、削除したりすることもできます。(左側の [インプット] [アウトプット] の欄に、直接入力することで、値を変更することもできます。) [OK] をクリックします。

これは、一時的な値であって、実行後は元に戻ります。


- シミュレーションの実行  をクリックします。
- 「うさぎ頭数」をクリックし、グラフ  をクリックします。

以下のように最後の「うさぎ頭数」が減少しているか、あるいはもっと異なるグラフが表示されます。下のグラフは、表関数の値を大きくした結果です。



## 8.8 表関数の変数

先ほどのモデルでは、うさぎ頭数の密度とその死亡の非線形な関係を示す補助変数を作成しました。しかし表関数だけの変数を作成した方がよいこともあります。特に、同じ表関数を2つ以上で使用する場合に有効です。

- [ファイル]メニュー→[別名で保存]を選び、*rabbit2.mdl* を、新しい名前 *rabbit3.mdl* で保存します。
- 変数  をクリックし、新しい変数「うさぎ急増による死亡への効果関数」を作成します。
- 「うさぎ急増による死亡への効果関数」から「うさぎ急増による死亡への効果」への矢印を描きます。
- 方程式 *f(x)* で「うさぎ急増による死亡への効果関数」をクリックします。
- [タイプ] のドロップダウンボックス [▼] から、[表関数] を選びます。[As graph] をクリックし、グラフ編集ダイアログを開き、先ほどと同じように、表関数の値を入力します。
- グラフ編集ダイアログを閉じ、[単位] に「Dmnl」と入力し、[OK] をクリックして方程式編集ダイアログを閉じます。方程式は以下のようになります。


「うさぎ急増による死亡への効果関数」 =  $(([0,0)-(4,2)], (0,0.9), (1,1), (2,1.2), (3,1.5), (4,2))$

Units : Dmnl

- 「うさぎ急増による死亡への効果」の方程式編集ダイアログを開きます。
- [タイプ] の2段目に表示されている [with Lookup] にあるドロップダウンボックス [▼] をクリックし、[標準] を選びます。方程式編集ボックスの下段がなくなりました。方程式の先頭をクリックし、[変数] リストから「うさぎ急増による死亡への効果関数」を選びます。また「標準化された頭数」に括弧 ( ) を付けます。

うさぎ急増による死亡への効果 = うさぎ急増による死亡への効果関数 (標準化された頭数)

➤ [ OK ] をクリックし、ダイアログを閉じます。

このモデルは、表関数を明示的に持つ変数が加わったことを除けば、*rabbit2.mdl* とまったく同じです。この方法で表関数の変数を作成することは、ここで記述したような簡単なモデルではあまり意味がありませんが、複雑な条件のもとでは大変役に立ちます。特に、同じ非線形関係を何度も利用する場合で有効です。シミュレーションの準備  をクリックした時、表関数の変数が強調表示され、それを変更することができます。

## 第 9 章





# 複数の表示画面

モデルサイズが大きくなると、それを理解することが難しくなります。こうしたモデルを 1 カ所に作図すると、圧倒されてしまいます。これを解決するには、表示画面（ビュー）と呼ばれるパーツに分けます。表示画面は、すべての Vensim のラインアップで利用できますが、Vensim PLE の古いバージョンは対応しません。

### 9.1 表示画面の働き

規模の大きいモデルは、1 つの表示画面よりも、複数の表示画面の方が、よりわかりやすくなります。表示画面は、本のページと同じようなもので、ページは物語の一部を構成しています。表示画面はスケッチを表示し、さらに変数や代行変数を通じて、他の表示画面と関連しています。複数の表示画面は、モデルを部門で分けることを可能にし、例えば生産、会計、顧客などに整理できます。

#### 9.1.1 グラフィックモデルと変数

Vensim のモデルは、方程式やテキスト形式で定義されます。図で示されたモデルは、すべての変数とすべての変数間の関係を表しているわけではありません。新しいスケッチを作成した時、表示画面に変数を加えます。これらの変数は、表示画面から削除しても、モデルに存在させることが可能です。変数とその構造は、上層に戻す  や下層に隠す  を使って非表示にすることができます。変数は方程式  $f(x)$  を使って、直接モデルに追加することができます（Vensim Professional と DSS では、テキストエディタ  を使うこともできます）。これらの変数は、変数  を使用して追加するか、ある変数の原因となるため Vensim が自動的に追加しない限り、表示されることはありません。

図で示されたモデルでは、同じ変数を別の表示画面に表示できます。反対に、どの表示画面にも表示されない変数もあります。変数は原因となる変数を持てる通常の変数か、原因となる変数を持たない代行変数があります。

**注意** 表示画面ごとに、関連しないモデルを作成することが可能です。しかし将来、それらを因果関係で結び付けたいと思わないのであれば、お勧めできません。新しいモデルを作成するほうが良いでしょう。


## 9.2 顧客の増加モデル

このモデルは、ある製品の潜在的な顧客が、すでにその製品を購入した顧客からの口こみによって、その製品の購入に至る課程を示しています。最初の表示画面は、普及の過程を示しています。第2の表示画面は、売られた製品の量を潜在的に制限する生産能力の変数を追加しています。第3の表示画面は、製品の販売によってもたらされる販売収入を示します。


この種類のモデルにおける理論的な背景は、モデリングガイドの4章にあります。

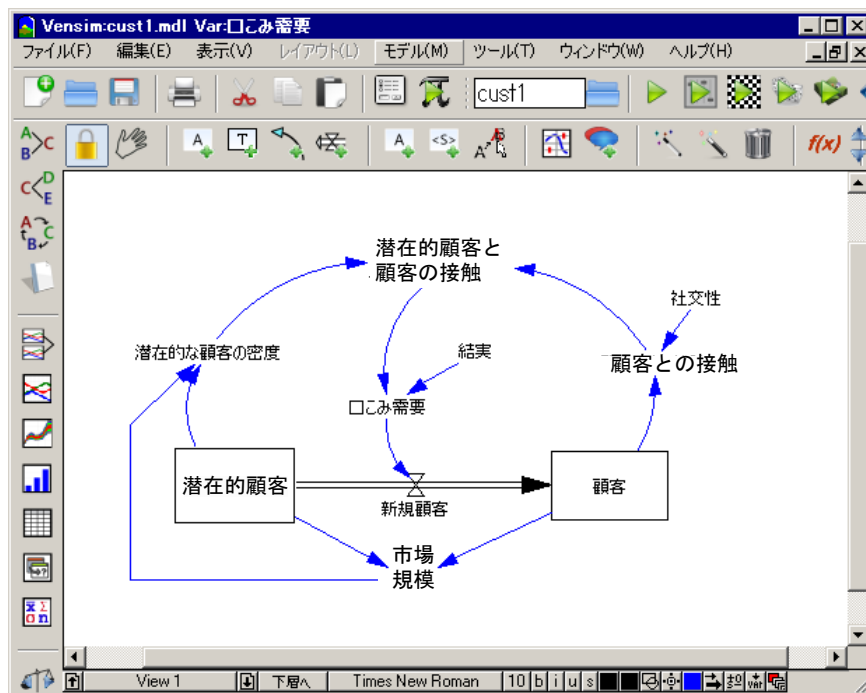
### 9.2.1 顧客の増加モデルの作成 (cust1.mdl)

英語版です。

- 既存のモデルを開きます。開く  をクリックし、*guide\chap09\complete* から、*cust1.mdl* を開き、*guide\chap09* に、*cust1.mdl* として保存します。「モデルのシミュレーション」に進みます。

または、




- 新しいモデルを作成します。新規モデル  をクリックするか、[ファイル] メニュー→[新規モデル] を選択します。
- デフォルトの[時間の範囲]で[OK]をクリックします。
- 下の図と同じモデルを作成します。
- *guide\chap09* に、*cust1.mdl* として保存します。




このスケッチには、ポリラインと呼ばれる矢印が含まれています。ポリラインは、矢



印が弧ではなく、線分の連続として表示されます。それ以外は、通常の矢印と同じです。Vensim PLE 及び PLE Plus では、ポリラインを描く機能はありません。したがって通常の矢印を利用します。ポリラインの機能を利用するには、Vensim Professional または DSS が必要です。

- 矢印  を、右クリックするか、**[Ctrl]**を押しながらクリックします。
- [タイプ] の [Polyline] にチェックを入れ、**[OK]** をクリックします。アイコンが、 に変わり、マウスポインタも変わります。
- 「マーケット」をクリックし、左に移動し、再度クリックします。次にまっすぐ上に移動し、もう一度クリックし、最後に「潜在的な顧客の密度」をクリックします。
- 矢印  を右クリックし、[タイプ] を [Normal] に戻します。

### 9.2.2 方程式の入力

- 方程式  をクリックし、それぞれの変数に、以下の方程式と単位を入力します。

新規顧客 = 口こみ需要  
Units : 人/月

潜在的顧客と顧客の接触 = 顧客との接触 \* 潜在的顧客の密度  
Units : 接触/月

顧客との接触 = 顧客 \* 社交性  
Units : 接触/月

顧客 = INTEG(新規顧客, 1000)  
Units : 人

結実 = 0.01  
Units : 人/接触

潜在的な顧客の密度 = 潜在的な顧客 / 市場規模  
Units : dmm1

潜在的な顧客 = INTEG( - 新規顧客, 1e+006)  
Units : 人

1e+006 とは、 $1 \times 10^6$  の 6 乗のこと。つまり  $1 \times 10^6$

社交性 = 20  
Units : 接触/人/月

市場規模 = 顧客 + 潜在的な顧客  
Units : 人

口こみ需要 = 非顧客と顧客の接触 \* 結実  
Units : 人/月

### 9.2.3 構文と単位のチェック


モデルをシミュレーションする前に、方程式と単位をチェックします。

- [モデル] メニュー → [モデルチェック] を選択します。または **[Ctrl]+[T]** を押します。[モデル O.K. です] と表示されます。



間違いが見つかった場合、モデルの構造が、上記の図と同じかどうか確認してください。構造が同じなら、それぞれの変数に対して、方程式編集ダイアログを開いて、入力した方程式がリストと同じかどうか確認してください。

- [モデル]メニュー [単位チェック]を選択します。または $\text{Ctrl} + \text{U}$ を押します。[単位 O.K. です]と表示されます。



エラーが表示された場合、どの変数に間違いがあるのかを確認します。問題のある変数の方程式編集ダイアログを開いて、先ほどのリストの単位と見比べます。

- 保存  をクリックするか、[ファイル]メニュー [保存]を選択します。もしくは $\text{Ctrl} + \text{S}$ を押して、モデルを保存します。

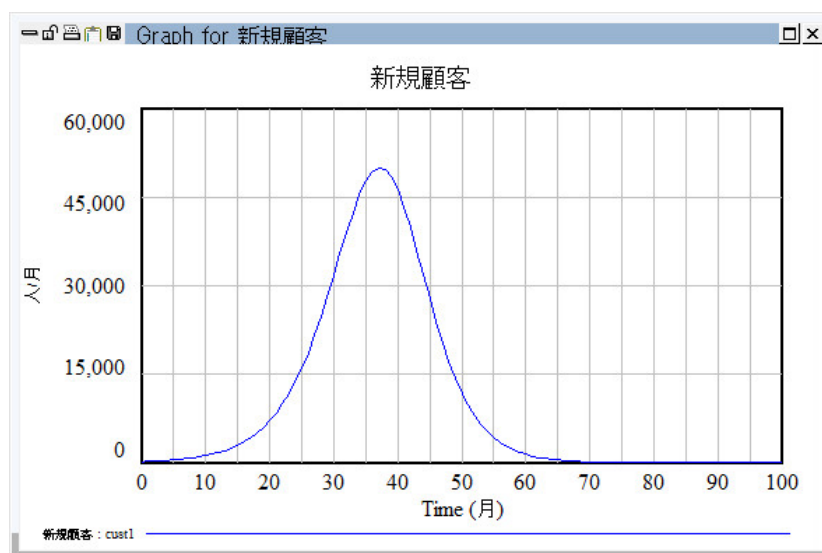
#### 9.2.4 モデルのシミュレーション

- ツールバーにあるシミュレーション名の編集ボックス  をクリックし、最初のシミュレーション実行名を「cust1」と入力します。
- シミュレーションの実行  をクリックします。

#### 9.2.5 モデルの分析

- グラフ  または、直接原因グラフ  を使って、「潜在的な顧客」や「顧客」、「新規顧客」などの主要な変数の振る舞いを見ます。

「新規顧客」は、次のようなグラフになります。



#### 9.2.6 表示画面の名称とモデルの保存

次の節を始める前に、表示画面に名前を付け、新しい名前モデルを保存します。





- [表示]メニュー [名前変更]を選択し、名前を「顧客」と入力し、[OK]をクリックします。
- [ファイル]メニュー [別名で保存]を選択し、名前を「*cust2.mdl*」と入力し、[保存]をクリックします。

### 9.3 表示画面「能力」の追加 (cust2.mdl)

ここでは、生産能力に関する検討が行えるようにモデルを拡張します。顧客からの需要は、製品を供給する能力を上回るかもしれません。そこで、表示画面「顧客」と関連を持ち、生産能力が限界に達した場合、顧客需要を制限する表示画面「能力」を作成します。

- [表示]メニュー [新規ファイル]を選択します。

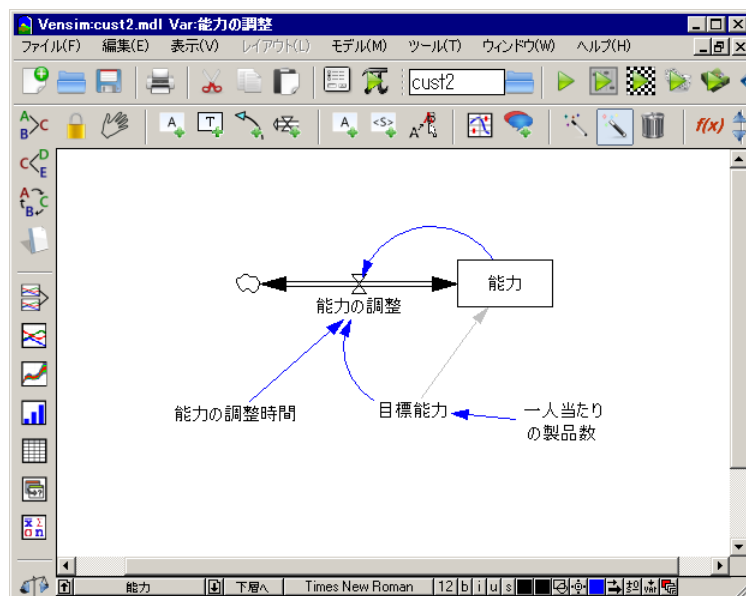
この表示画面に名前を付けます。これによって、表示画面を簡単に移動できます。

- [表示]メニュー [名前変更]を選択します。名前を「能力」と入力し、[OK]をクリックします。
- スtock変数  を選択し、スケッチ上でクリックします。名前を「能力」と入力し、を押します。
- フロー変数  を選択し、「能力」の左でクリックし、次に「能力」の上をクリックします。名前を「能力の調整」とします。
- スケッチ移動  をクリックし、次に「能力の調整」の左側にあるパイプ上のハンドル（小さな円）を右クリックします。矢印のオプションダイアログが開きます。  
[矢頭]にチェックを入れ、[OK]をクリックします。


この追加の矢頭は、このフローが双方向に作用することを示しています。つまり、フロー「能力の調整」は、「能力」を増加させることも減少させることもできます。しかし、フローの実際の振る舞いは、追加した矢頭によってではなく、方程式によって決まることに注意してください。

- 他の変数と、それらと関連する矢印を、以下のように作成します。



この図のように表示されない時は、[モデル]メニュー [設定]の[スケッチ]タブにある[Show initial causes on model diagrams]【図に初期値を表示する】や、[Show <shadow> variables on model diagrams]【図に代行変数を表示する】などで変更できます。



### 9.3.1 代行変数ツール


代行変数  を選択し、「目標能力」の下をクリックします。ダイアログが開きます。





- 変数のリストから「口こみ需要」を選択するか、「口こみ需要」が、リストの中で強調表示されるまで、始めの文字列を入力します。を押すか、「OK」をクリックします。
- 矢印  を選択し、「口こみ需要」をクリックし、「目標能力」をクリックします。

表示画面「能力」が完成しました。「口こみ需要」という変数により最初の表示画面「顧客」と関連を持ちます。

**注意** 代行変数は、他の変数に変化をもたらすためだけに用いられます。もし、別の変数から代行変数に矢印を結ぼうとしても、矢印はつながりません。別の表示画面にある変数を、ある表示画面の変数により影響を受けるようにするためには、モデル変


数  を使います。

モデル変数  は、変数  とは異なります。

### 9.3.2 方程式の追加

- 方程式  を選択します。

すでに方程式の定義を持つ「口こみ需要」以外の、すべての変数が反転表示されます。

- 方程式  をクリックし、この表示画面のそれぞれの変数に、以下の方程式と単位を入力します。

能力 = INTEG (-能力の調整, 目標能力)  
Units : 個/月



能力の調整 = (目標能力-能力)/能力の調整時間  
Units : 個/月/月

目標能力 = 口こみ需要\*一人当たりの製品数  
Units : 個/月

能力の調整時間 = 12  
Units : 月




一人当たりの製品数 = 1  
Units : 個/人

次に、表示画面「顧客」に戻り、その構造に、表示画面「能力」にある変数を関連づけ、フィードバックループを完成させます。

- ステータスバーの、モデル構築ウィンドウを選択する  View 1  の [能力] と表示された欄をクリックし、[顧客] を選択します。

名前は「View1」ではなく、「顧客」となっています

注意 キーボードの [PgUp] や [PgDn] を使って、表示画面を切り換えることもできます。

- 代変変数  を選択し、「新規顧客」の右下をクリックします。変数のリストから「能力」を選択するか、「能力」が、リストの中で強調表示されるまで、始めの文字列を入力します。  を押すか、[OK] をクリックします。
- 「一人当たりの製品数」に対しても、これと同じことを繰り返します。
- 必要に応じて、「マーケット」と矢印を動かし、スペースを空けます。
- 矢印  を選択し、「能力」と「新規顧客」を結びます。次に、「一人当たりの製品数」と「新規顧客」を結びます。

### 9.3.3 方程式の変更

- 方程式  を選択します。

「新規顧客」だけが反転表示されています。これは、この表示画面の中で、原因となる変数を追加したのがこの変数だけだからです。製品の販売は、工場の生産能力に制限されます。したがって、「口こみ需要」もしくは、工場の生産能力を一人の顧客当たりの製品数で割った値（「能力」/「一人当たりの製品数」）よりも小さい値を返す方程式を入力します。


- 「新規顧客」をクリックし、方程式を以下のように変更します。

新規顧客 = MIN(口こみ需要, 能力/一人当たりの製品数)  
Units : 人/月

[関数] タブのリストから、[MIN] を選択するか、直接入力します。


## 9.4 表示画面「販売収入」の追加

製品の売り上げによる販売収入とその累積収入の表示画面を追加します。




- [表示] メニュー [新規ファイル] を選択します。
- [表示] メニュー [名前変更] を選択し、「販売収入」と入力し、を押します。

## 9.5 モデル変数ツール

VensimPLE 及び PLE Plus を使っている場合、モデル変数ツールはありません。したがって、「一人当たりの製品数」と「新規顧客」を代行変数として作成し、それ以外の変数を追加します。

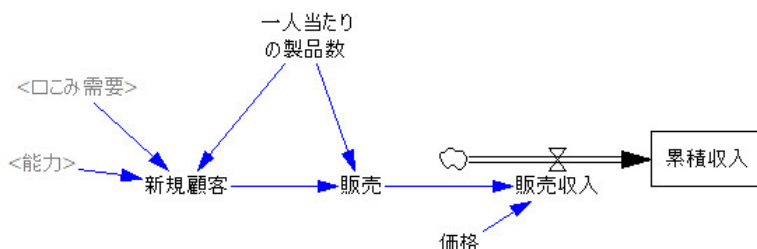
- モデル変数  を選択します。表示画面をクリックし、変数リストから [新規顧客] を選択します。

これで、表示画面に「新規顧客」とそれに影響する変数が追加されます。影響する変数は重なって表示されるので、それらを再配置します。

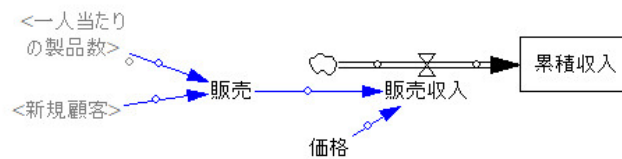
- スケッチ移動  (もしくは今選択されているモデル変数 ) を使って、「口こみ需要」や「能力」を「新規顧客」の左に配置します。
- モデル変数  を選択します。「一人当たりの製品数」をクリックします。

これで、代行変数を通常の変数に変更しました。「一人当たりの製品数」は定数であるため、それに影響を及ぼす変数は追加されません。

- 以下のように、変数と矢印を追加します。



Vensim PLE 及び PLE Plus の場合は、以下のような図になります。



### 9.5.1 方程式の追加

- 方程式 をクリックし、この表示画面に反転表示された変数に、以下の方程式と単位を入力します（他の変数は、すでに方程式が入力されています）。

販売 = 新規顧客 \* 一人当たりの製品数  
Units : 個/月

価格 = 50  
Units : 円/個

販売収入 = 販売 \* 価格  
Units : 円/月

- 「累積収入」の方程式を入力する際、[ 補充的に使用 ] にチェックを入れます。これは、この変数が、他では使用されないことを示します。

累積収入 = INTEG (販売収入, 0)  
Units : 円

- [ ファイル ] メニュー [ 保存 ] を選択するか、 を押して、モデルを保存します。

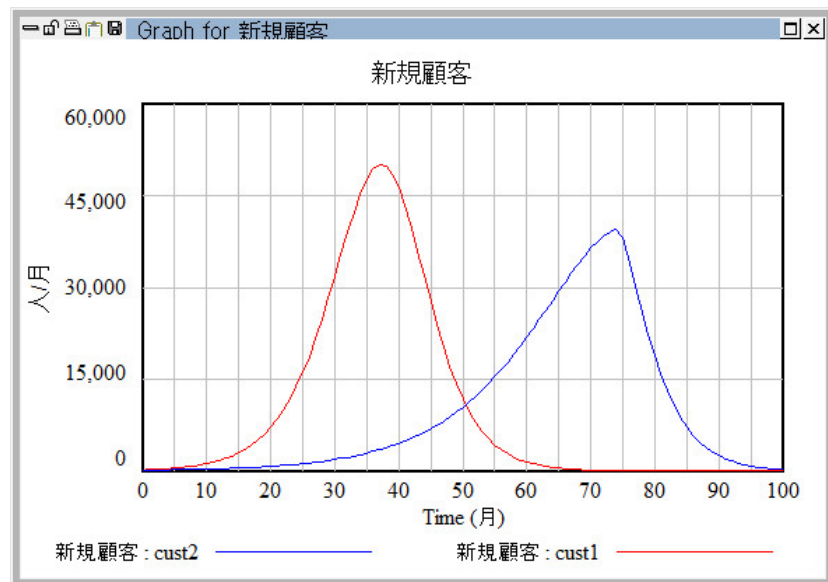
### 9.5.2 モデルのシミュレーション

- [ モデル ] メニュー [ モデルチェック ] を選択するか、 を押して、モデルに間違いがないかチェックします。
- [ モデル ] メニュー [ 単位チェック ] を選択するか、 を押して、単位に矛盾がないかチェックします。
- データセット名 cust2 でシミュレーションを実行します。

### 9.5.3 モデルの分析

- グラフ または、直接原因グラフ を使って、「潜在的な顧客」や「顧客」、「新規顧客」などの主要な変数の振る舞いを見ます。もし、自身で作成したのではなく、既存の cust2.mdl を用いた場合、表示される結果は、現在のデータセット (cust2) だけです。以前の結果も表示する場合には、制御パネル の [ データセット ] タブから「cust1」を読み込みます。
- 「新規顧客」をワークベンチ変数とし、グラフ をクリックします。

注意 データセット *cust1* は、*cust1.mdl* のすべての変数の結果を持っています。しかし新たに作成されたモデルに固有の変数結果はありません。元のモデルにはなかった変数をグラフ表示しようとすると、*cust2* の結果しか見ることができません。



2つのシミュレーションの実行結果の違いがわかります。生産能力の制約によって、2つ目のシミュレーションは、販売の増加が緩やかになり、そのピークも遅れます。また販売期間は長くなっています。

#### 9.5.4 モデルの保存

次の節を始める前に、モデルを新しい名前で作成します。

- [ファイル]メニュー [別名で保存]を選択し、名前を「*cust3*」とし、[OK]をクリックします。
- [ファイル]メニュー [閉じる]を選択し、モデルを閉じます。

### 9.6 詳細な能力モデル (*cust3.mdl*)








より正確に、企業の生産能力部門を設計してみます。今のモデルより詳しく能力部門をあらわす既存のモデルがあれば、今のモデルを改良するために、このモデルを用いることができます。投資に対する建築物の遅れを考慮した建設能力と老朽化による能力低下を考慮するモデルとして *cap1.mdl* があります。

本節では、2つの異なるモデル（構造と方程式が異なる）を1つのモデルに合体させる方法を紹介します。



### 9.6.1 コピーと貼り付け


別にある既存のモデル (*cap1.mdl*) の構造を利用します。このモデルには、能力を決定づける改良された方程式などがあります。しかしながら、顧客需要に関する方程式はなく、その代わりに「求められる生産量」を決定する STEP 関数を利用しています。STEP 関数については、7 章の解説を参照してください。このモデル構造をこれまでのモデルに貼り付け、さらにすでにある変数と *cap1.mdl* からもたらされた変数を関連づけます。

- 開く  をクリックし、*guide\chap09* から *cap1.mdl* を開きます。
- [編集] メニュー [全て選択] を選びます。もしくは、**Ctrl**+**A** を押すか、スケッチ移動  をクリックし、マウスのドラッグですべての構造を囲みます。
- コピー  をクリックします。もしくは、[編集] メニュー [コピー] を選択するか、**Ctrl**+**C** を押します。
- 開く  をクリックするか、[ファイル] メニュー [開く] を選択し、*cust3.mdl* を開きます。もしくは、[ファイル] メニューの最近使用したファイルのリストから、*cust3.mdl* を選択します。
- 表示画面が、「能力」でない場合は、ステータスバーのモデル構築ウィンドウを選択する  **View 1**  をクリックし、「能力」を選択します。
- 貼り付け  をクリックします。もしくは、[編集] メニュー [貼り付け] を選択するか、**Ctrl**+**V** を押します。


貼り付けモードの選択ダイアログが開きます。

1. Replicate【複製】: モデルの表示画面に、構造や方程式を貼り付けます。このとき、同じ名前ですでに存在する変数は改名されます。
  2. Picture【画像】: モデルの表示画面に、構造だけを貼り付け、方程式は貼り付けられません。現在のモデルに存在しない変数をモデルが貼り付けられた場合、警告のダイアログが開きます。
- 変数と方程式をモデルに追加するために、Replicate【複製】を選択します (デフォルトで選択されています)。[OK] ボタンをクリックします。

新しい構造が、表示画面上に、コピー元と同じ位置で貼り付けられます。

- スケッチ移動  をクリックします。点線で強調表示された箱の中央で、マウスを押して、マウスを押したまま、既存の構造の下に移動させます。
- 箱の外側をクリックし、スケッチ上に変数を固定します。

**注意** 移動する前に、箱の外側をクリックした場合、新しい構造は、今ある構造の一番上に貼り付けられます。これを元に戻すには、[編集] メニュー [元に戻す] を選び、張り付けた構造を消去します。その後、改めて Replicate【複製】により貼り付けます。


- 変数合体  を選択し、新しい変数「能力 0」の上でマウスを押し、ドラッグしながら「能力」の上に移動させ、マウスボタンを離します。

[ この操作は変数“能力”を“能力 0”と入れ替えます。続行しますか? ] というダイアログが表示されます。









- [ はい ] をクリックします。
- 「能力 0」の上でマウスを押し、ドラッグしながら「能力 0」を元の位置（「投資」と「縮小」の間）に移動させ、マウスを離します。

変数合体では、*cap1.mdl* にあった「能力 0」の全ての因果関係と方程式を「能力」に置き換えます。そのため、「能力」の方程式と因果関係はなくなります。しかし、「能力」の入力値として使用された変数（「能力の調整」と「新規顧客」）は、「能力 0」の入力値として使用されています。

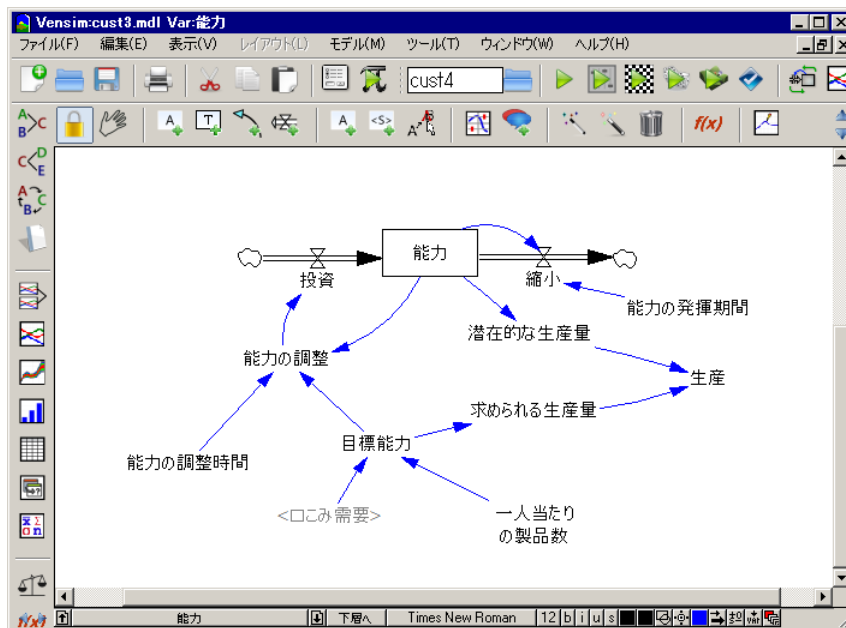
新しい表示画面「能力」の構造では、「求められる生産量」を決める変数が必要であり、それらは「口こみ需要」と「一人当たり製品数」から得られます。これまでの「能力」の構造では、この計算は変数「目標能力」で計算されていました。そこで、この方程式をそのまま維持し、古い変数「目標能力」を新しい変数「目標能力 0」に合体させます。

- 変数合体  を使って、「目標能力」を「目標能力 0」の上にドラッグします。警告メッセージが出たら、[ はい ] をクリックします。

「能力の調整時間」も同じように元の値を維持します。

- 変数合体  を使って、「能力の調整時間」を「能力の調整時間 0」の上にドラッグします。警告メッセージが出たら、[ はい ] をクリックします。
- 「口こみ需要」と「一人当たりの製品数」を、「目標能力」の下にドラッグします。
- 消去  を選択します。「能力の調整」をクリックし、モデルから消去します。フローがあったところの左にあった雲をクリックします。
- スケッチ移動  を使って、変数と矢印を移動し、スケッチを整えます。
- もし、矢印のハンドルが見つからず、移動できない時は、矢印  を選択し、変数の間に新しい矢印を作成します。"This connection exists, remove the old arrow?" 【この結合はすでに存在します。古い矢印を消去しますか? 】という警告メッセージが出たら、[ はい ] をクリックします。
- 変数  を選択します。「能力 0」をクリックし、名前を示す編集ボックスを開きます。名前の末尾の「0」とスペースを削除し、 を押します。
- 「能力の調整 0」をクリックし、名前の末尾の「0」とスペースを削除し、 を押します。
- 矢印  を選択します。「目標能力」と「求められる生産量」を結合します。（下図のようになります）

こうしたスケッチに対する操作により、必要となる方程式を維持しつつ、適切な変数名となりました。表示画面は、以下のようになります。



- 方程式  $f(x)$  を選択してください。「求められる生産量」をクリックし、「目標能力」に変更します。

求められる生産量 = 目標能力  
Units : 個/月

反転表示された方程式はなくなりました。もしまだ残っている場合、その変数をクリックし、以下の方程式のリストと照合し、必要ならば変更してください。

- 「生産」をクリックし、[ 補充的に使用 ] にチェックを入れます。

### 9.6.2 表示画面「能力」の方程式

能力 = INTEG (投資 - 縮小, 目標能力)  
Units : 個/月

能力の調整 = (目標能力 - 能力) / 能力の調整時間  
Units : 個/月/月

能力の発揮期間 = 20  
Units : 月

求められる生産量 = 目標能力  
Units : 個/月

投資 = 能力の調整  
Units : 個/月/月

潜在的な生産量 = 能力  
Units : 個/月

生産 = MIN(求められる生産量, 潜在的な生産量)  
Units : 個/月

縮小 = 能力/能力の発揮期間  
Units : 個/(月\*月)

目標能力 = 口こみ需要 \* 一人当たりの製品数  
Units : 個/月

能力の調整時間 = 12  
Units : 月

### 9.6.3 単位同義語

単位同義語は、異なる名前を同じ単位とみなすものです。単位を入力する際、英語の場合、単数形と複数形のどちらも使用することがあります。また日本語の単位でも、「個」と「個数」が混在する場合も考えられます。単位チェックでは、それらが同義であることを定義しておかなければ、異なる単位とみなします。

いくつかの同義語がすでに定義されています。例えば、Month と Months や Year と Years などがあります。

➤ [モデル]メニュー [単位チェック]を選びます。もしくは $\text{Ctrl} + \text{U}$ を押します。

単位エラーが表示されます。それは、英語のモデル *cap1.mdl* を導入したため、Gadget【部品】と Month【月】という日本語化した単位と異なるものが存在するためです。すべての方程式から、「Gadget」を見つけ出し、それを「個」に書き換えることもできます。しかし、「Gadget」と「個」は同じ単位であるので、同義語として定義します。さらに、複数形で入力した場合に備えて、複数形の (Gadgets) も追加します。

➤ [モデル]メニュー [設定]を選択し、[単位同義語]タブをクリックします。

➤ 編集ボックスの中に、「Gadget,Gadgets, 個」と入力し、[編集して追加]をクリックします。

➤ リストにある「Month,Months」をクリックし、[選択して修正]をクリックします。編集ボックスを「Month,Months, 月」のようにし、[編集して追加]をクリックします。[OK]をクリックします。


➤ [モデル]メニュー [単位チェック]を選択します。もしくは $\text{Ctrl} + \text{U}$ を押します。


単位のチェックができました。エラーが表示された場合、エラーの内容を見て、解決します。それぞれの変数の単位は、先ほどのリストを参照してください。

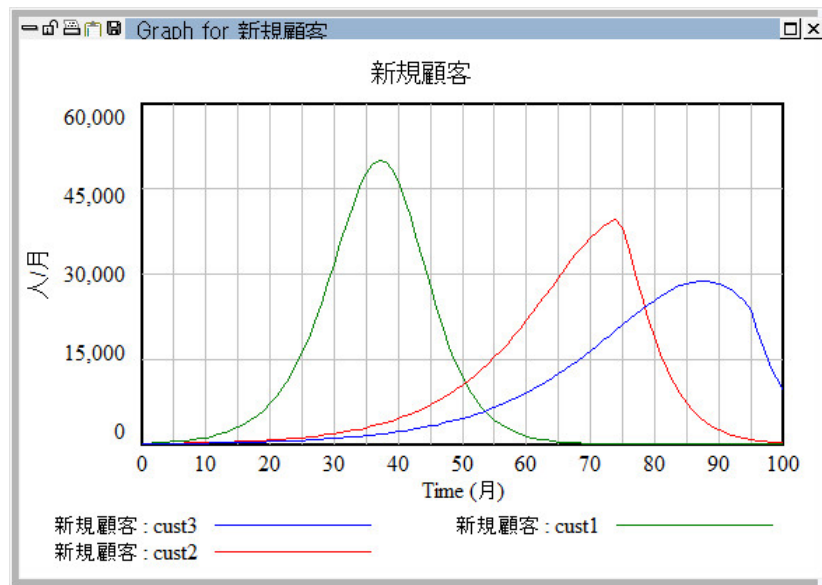
## 9.7 モデルのシミュレーションと分析

➤ [モデル]メニュー [モデルチェック]を選びます。もしくは $\text{Ctrl} + \text{T}$ を押します。

➤ *cust3* をシミュレーションします。

➤ 制御パネル  をクリックし、[データセット]タブをクリックします。*cust1* と *cust2* が読み込まれていなければ、読み込みます。





➤ 「新規顧客」を選択し、ワークベンチ変数とします。グラフ  をクリックします。

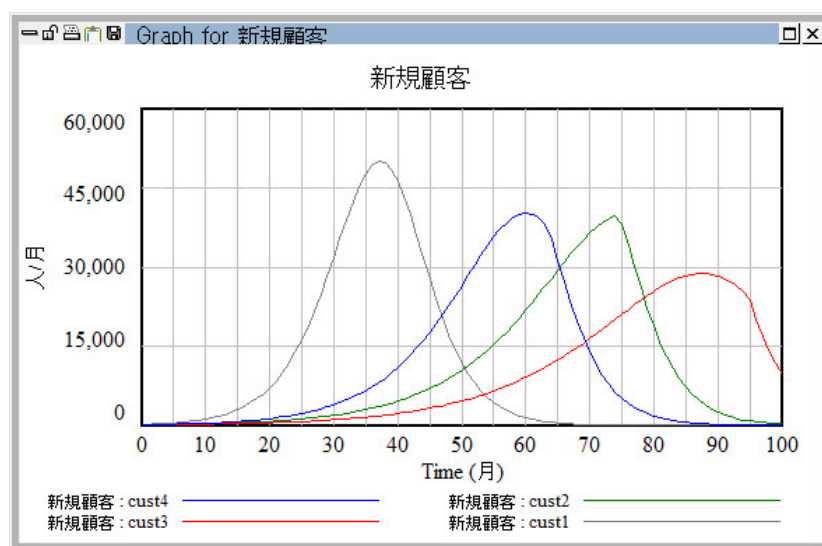


*cust3* の「新規顧客」の曲線が、他のシミュレーション結果に比べ、緩やかで、ピークが遅く、終りが延びていることがわかります。この振る舞いは、定年退職などによる生産能力の減少により、*cust3.mdl* において生産能力の制約があるためです。

## 9.8 生産能力の投資戦略

新しい市場の供給では、競争相手よりも製品の供給が遅れるために、市場占有率（「新規顧客」と「顧客」）を失うことがあります。より早く製品を供給するには、どのような方策があるでしょうか。「能力の調整時間」を減らすことで、生産能力を高めます。

- シミュレーション名の編集ボックス  をクリックし、シミュレーション実行名を「*cust4*」と入力します。
- シミュレーションの準備  をクリックします。
- 表示画面 [能力] で、強調表示された「能力の調整時間」をクリックし、「4」と入力し、 を押します。
- シミュレーションの実行  をクリックします。
- 「新規顧客」を選択し、ワークベンチ変数とします。グラフ  をクリックします。



「新規顧客」が理想とする最初のモデル *cust1.mdl* に近づきました。しかし生産能力の制約は解消されたわけではなく、押し戻されているに過ぎません。

## 第 10 章

# 出力のカスタマイズ


分析ツールの設定 (Vensim PLE と PLE Plus は対応しません) 及び、カスタムグラフとカスタム表の作成によって、出力をカスタマイズできます。

### 10.1 分析ツールからの出力

Vensim PLE と PLE Plus を除き、すべての分析ツールにおいて、出力の形式と内容をカスタマイズできます。


#### 10.1.1 樹形図

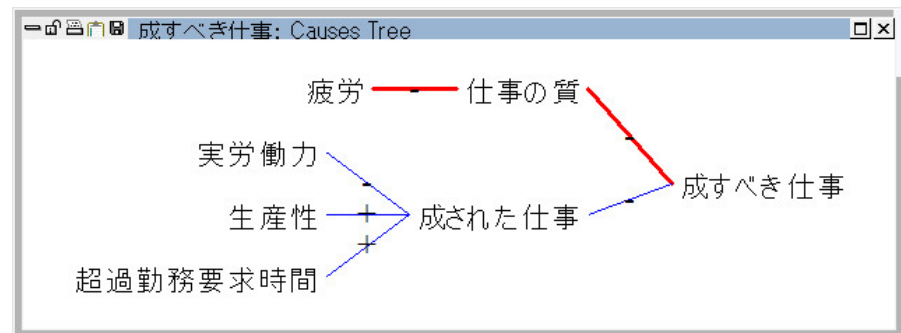
矢印の幅、色、極性などといった図の表示をカスタマイズすることができます。

- `guide\chap04\complete` から `project.mdl` を開きます。「成すべき仕事」をクリックし、ワークベンチ変数とします。
- 因果ツリー  を右クリックします。



これは英語版です。4 章で日本語版を作成しています。



- [属性]にある[色][幅][Polarity][極性]のチェックボックスにチェックを入れ、[OK]をクリックします。
- 因果ツリー  をクリックします。




### 10.1.2 グラフと直接原因グラフ


グラフは、異なる色や太さの線を用いて表示できます。グラフ  のオプションを操作します。同じ方法が直接原因グラフ  にも応用できます。


➤ 8 章で作成した *cust3.mdl* を開きます。

もしくは、

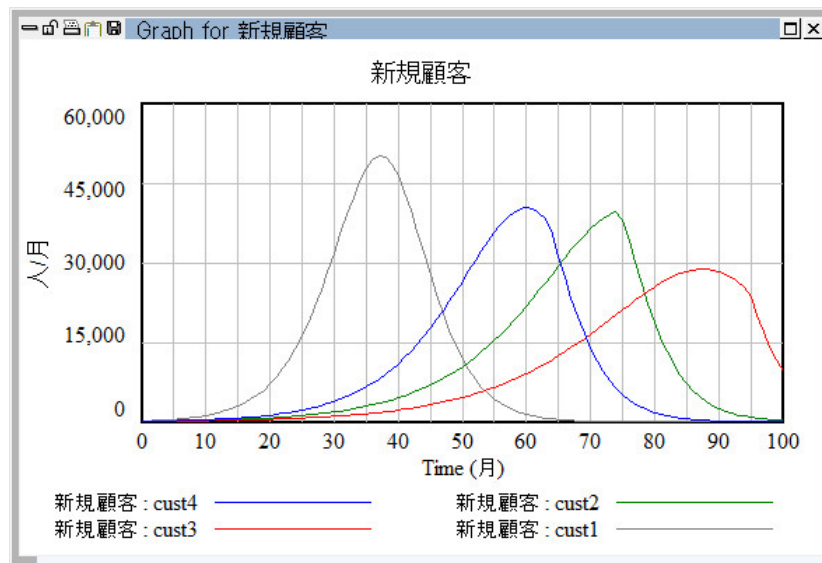
- *guide\chap08\complete* から *cust3.mdl* を開きます。
- 制御パネル  を開き、[ データセット ] タブをクリックします。*cust1* から *cust4* まだが全て読み込まれているか確認します。もしまだなら、左側の変数リストからダブルクリックで読み込みます。
- 表示画面「顧客」にある変数「新規顧客」をクリックし、ワークベンチ変数とします。

### 10.1.3 細い線 (カラー)

➤ グラフ  をクリックします。

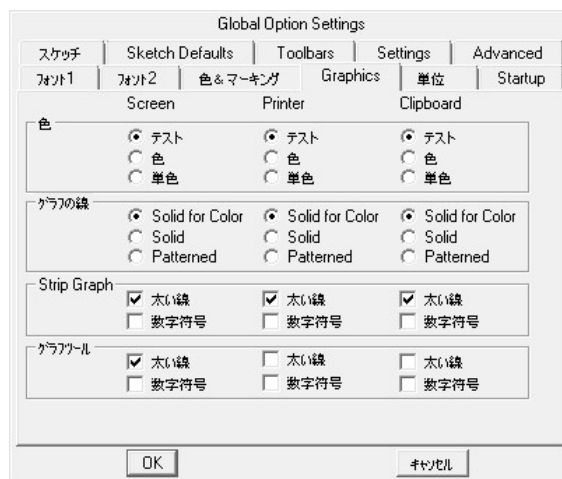
Vensim のグラフ  のデフォルトでは、異なる色の細い線が描かれます。






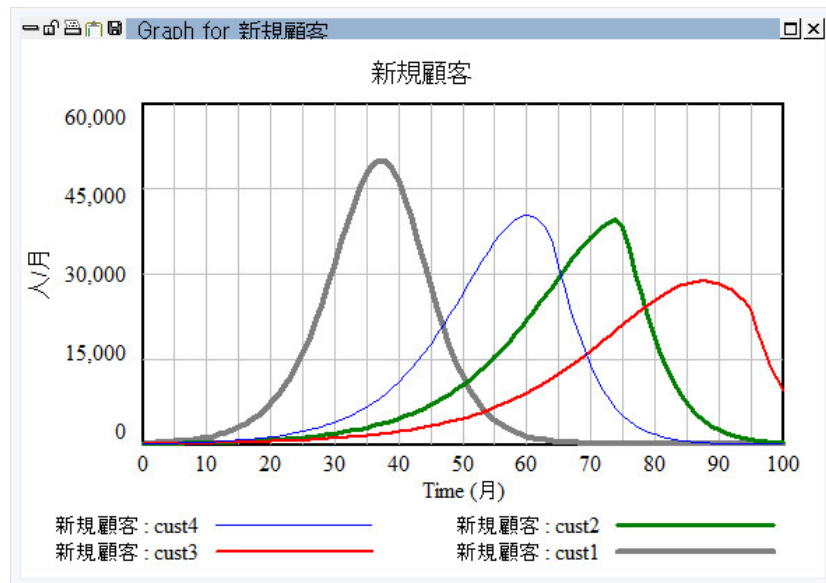
#### 10.1.4 太い線（カラー）

- [ ツール ] メニュー [ オプション ] を選び、[ Graphics ] タブをクリックします。  
[ グラフツール ] の一番左にある [ 太い線 ] にチェックを入れます。[ OK ] をクリックします。



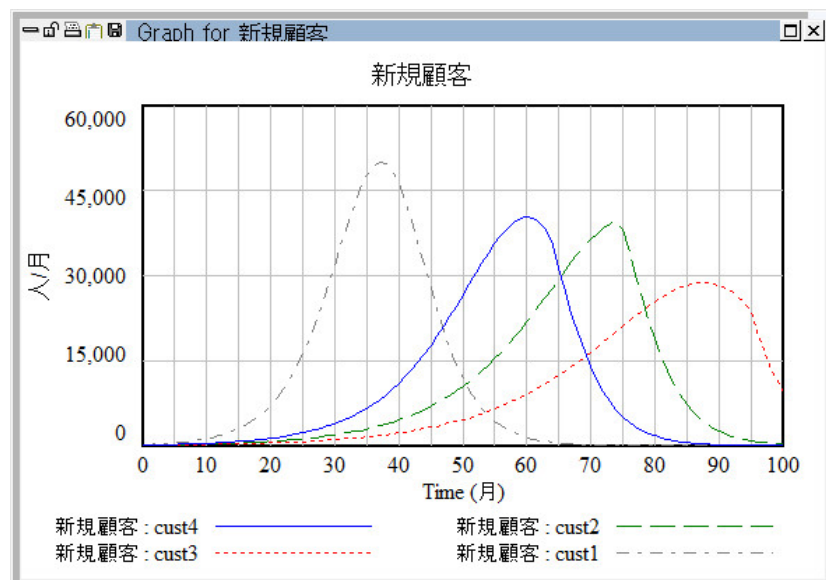
先に、グラフ  で出力されたグラフが、太い線に更新されました。グラフのオプションは、開いているすべてのグラフと、新しく作成したグラフに影響することに注意してください。

[ Graphics ] タブのダイアログは 3 列で構成されており、線の表示は [ Screen ] 【画面】( 左 ) [ Printer ] 【印刷】( 中央 ) [ Clipboard ] 【クリップボード】( 右 ) に対応しています。ここでは、[ Screen ] だけを変更します。



### 10.1.5 線の種類（カラー）

- [ ツール ] メニュー [ オプション ] を選び、[ Graphics ] タブをクリックします。  
[ グラフの線 ] の [ Screen ] 列にある [ Patterned ] 【模様】を選びます。[ OK ] をクリックします。







設定されたグラフィックオプションをデフォルトに戻します。

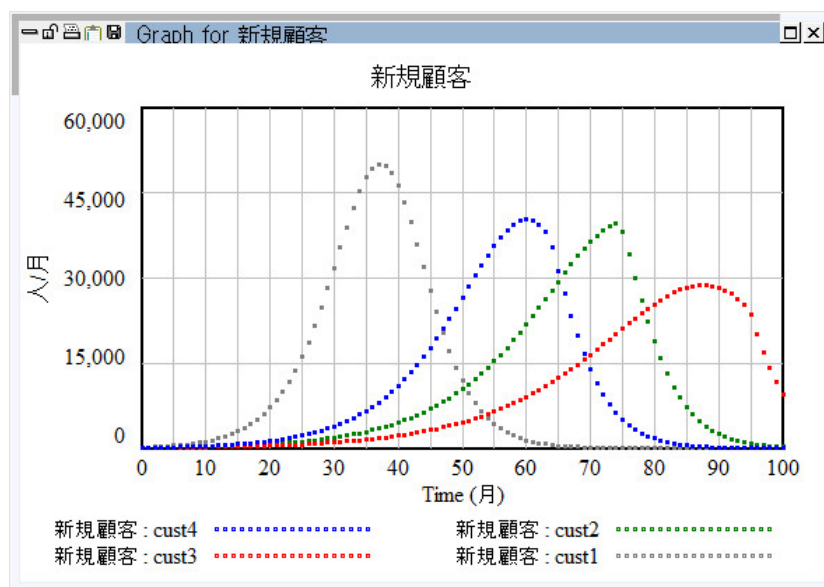
- [ ツール ] メニュー [ オプション ] を選び、[ Graphics ] タブをクリックします。  
[ グラフの線 ] の [ Solid for Color ] 【色塗りつぶし】を選びます。
- グラフを閉じるため、アウトプットウィンドウを閉じるボタンをクリックするか、

[Del]を押します。


### 10.1.6 点のみのグラフ

- グラフ  を右クリックします。[ Line Type ]【線の種類】の[ 点のみ ]を選択し、[ OK ]をクリックします。
- グラフ  をクリックします。

グラフのツールは、 から  に変更されます。



グラフツールのオプションを元に戻します。

- グラフ  を右クリックします。[ Line Type ]の[ Linear Interp ]【直線で補間】を選択し、[ OK ]をクリックします。

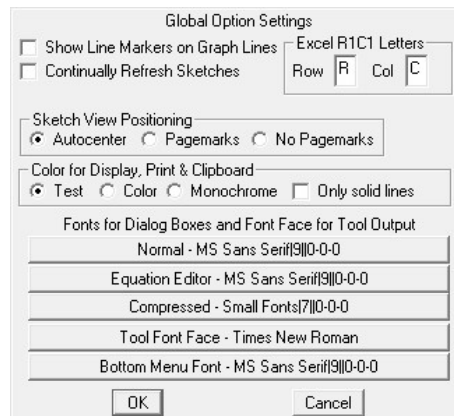
## 10.2 マーカーのついた線

線にマーカーを付けることもできます。Vensim PLE と PLE Plus では、簡単なオンとオフのスイッチがあり、その他のラインアップでは、カスタマイズすることができます。

マーカーは、グラフの線に沿って表示される文字です。デフォルトでは、数字や文字に設定されていますが、記号も使用できます。

Vensim PLE と PLE Plus では、

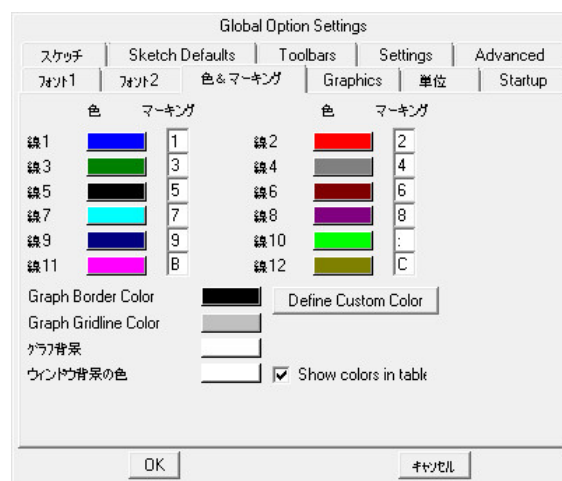
- [ オプション ] メニュー [ オプション ] を選びます。



- [ Show Line Markers on Graph Lines ] 【グラフの線にマーカーを表示する】にチェックを入れます。

Vensim professional と DSS では、


- [ ツール ] メニュー [ オプション ] を選び、[ 色&マーキング ] タブをクリックします。

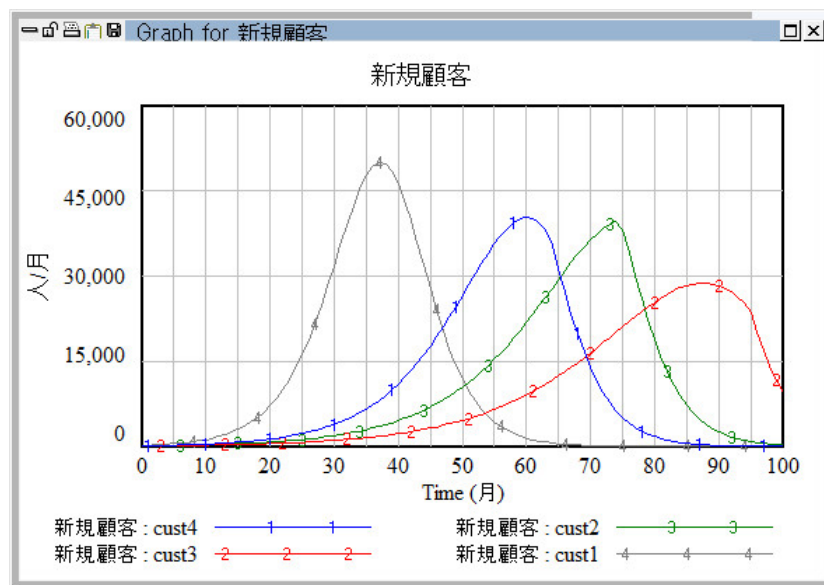


色とマーカーの対応関係が示されています。色を変更することが可能です。また、マーカーは別の 1 文字に変更することができます。

- [ Graphics ] タブをクリックします。[ グラフツール ] の左下にある [ 数数字号 ] にチェックを入れ、[ OK ] をクリックします。

すべてのラインナップで、


- グラフ  をクリックします。



[ 数数字号 ] のチェックをはずして、[ Global Options Setting ] のダイアログをデフォルトに戻します。

注意 [ Global Options Setting ] のダイアログで [ 色&マーキング ] や [ Graphics ] タブの設定を変更すると、グラフ表示と印刷結果が変わります。


## 10.3 カスタムグラフ

カスタムグラフは、グラフの内容をカスタマイズします。例えば、表示する変数を選択したり、シミュレーション結果を選択したり、その表示形式を変えることができます。カスタムグラフは、制御パネル  の [ グラフ ] タブから作成します。


*cust3.mdl* が開いていなければ、

- *guide\chap08\complete* から *cust3.mdl* を開きます。

カスタムグラフを作るために、

- ツールバーにある制御パネル  をクリックします。[ グラフ ] タブをクリックします。

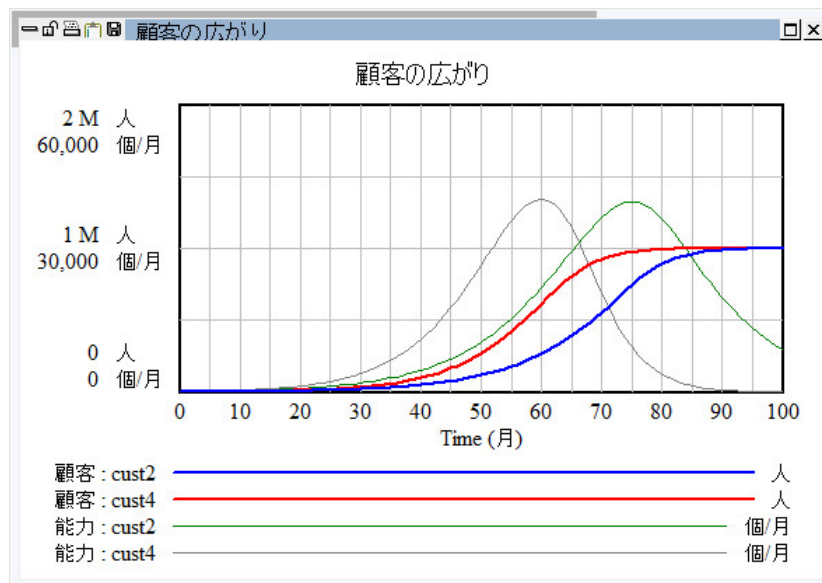



- [ 新規 ] をクリックします。カスタムグラフダイアログが開くと、[ タイトル ] の編集ボックスにカーソルがあります。
- タイトル名を「顧客の広がり」と入力します。
- マウスを、左下にある [ 変数 ] 近くに移動させ、一番上の [ 選択 ] をクリックします。変数のリストが表示されます。スクロールバーを移動させ、[ 顧客 ] をダブルクリックします。(または、「顧客」がリストの中で強調表示されるまで、始めの文字列を入力し、を押します。)
- 第 2 の [ 選択 ] をクリックし、「顧客」をダブルクリックします。
- 第 3 の [ 選択 ] をクリックし、「能力」をダブルクリックします。
- 第 4 の [ 選択 ] をクリックし、「能力」をダブルクリックします。
- 第 1 と第 2 の変数間の左側にある [ Scale ] 【目盛り】にチェックを入れます。
- 第 3 と第 4 の変数間の左側にある [ Scale ] にチェックを入れます。
- 第 1 の変数の右側にある [ データセット ] をクリックし、「cust2」と入力します。
- 第 2 の変数の右側にある [ データセット ] をクリックし、「cust4」と入力します。
- 第 3 の変数の右側にある [ データセット ] をクリックし、「cust2」と入力します。
- 第 4 の変数の右側にある [ データセット ] をクリックし、「cust4」と入力します。
- 第 1 の変数の右側にある [ LineW ] 【線の幅】をクリックし、「2」と入力します。
- 第 2 の変数の右側にある [ LineW ] をクリックし、「2」と入力します。

LineW のデフォルト値は 1 のため、その値が空白であれば幅「1」の線が表示されます。

カスタムグラフダイアログは、以下のようになります。

- [ OK ] をクリックします。カスタムグラフダイアログが閉じ、制御パネルの [ グラフ ] が残ります。
- 「グラフ」タブの「顧客の広がり」をクリックし、[ 表示 ] をクリックします。



グラフが表示されます。制御パネル  の [ グラフ ] タブの中の [ 修正 ] をクリックし、項目を修正することで、異なるオプションを試すことができます。もしカスタムグラフダイアログの変数名の横に、データセットが入力されていなくてもグラフは表示されます。表示されるグラフは、[ データセット ] タブにおいて、一番上に読み込まれているデータセットから作成されます。

## 10.4 カスタム表


カスタム表は、表の内容をカスタマイズするために使用されます。つまり、選択された変数やシミュレーションの結果を表示します。カスタム表は、カスタムグラフのように、

制御パネル  の [ グラフ ] タブで作成します。


*cust3.mdl* が開いていなければ、

- *guide\chap08\complete* から *cust3.mdl* を開きます。

カスタム表を作るために、

- ツールバーにある制御パネル  をクリックします。[ グラフ ] タブをクリックします。
- [ 新規 ] をクリックします。カスタムグラフダイアログが開くと、[ タイトル ] の編集ボックスにカーソルがあります。
- ダイアログの下方にある [ As Table ] 【表として作成】をクリックすると、カスタム表ダイアログが開き、[ タイトル ] の編集ボックスにカーソルがあります。




- [ タイトル ] に、「顧客の広がり表」と入力します。
- ダイアログの左下にある [ 変数 ] をクリックします。変数のリストが表示されます。スクロールバーを移動させ、「顧客」をダブルクリックします。(または、「顧客」がリストの中で強調表示されるまで、始めの文字列を入力し、を押します。)
- 変数名が入力された編集ボックスの右側にある [ 追加 ] をクリックします。
- 再度 [ 変数 ] をクリックし、変数リストから「能力」をダブルクリックします。
- 再度 [ 追加 ] をクリックします。
- カスタム表ダイアログは、以下のようになります。



- [ OK ] をクリックします。カスタム表ダイアログが閉じ、制御パネルの [ グラフ ] が残ります。
- 「顧客の広がり」をクリックし、[ 表示 ] をクリックします。

Time (月)	0	1	2
顧客	1,000	1,199	1,389
能力	199.80	189.81	192.78

選択した変数を含む表が表示されます。表示する変数を追加することは容易です。また、表の左端にある変数のラベル付けを行うこともできます。そのためには制御パネル  の [ グラフ ] タブの中の [ 修正 ] をクリックし、項目を修正することで、設定を変更します。もし表示されている変数の順番を変更したい場合は、[ 表コンテンツ・ドラッグで再配列 ] のリストの中で、その変数をドラッグします。また、コメント行を追加することもできます。また数値の表示形式として、C 言語のフォーマットを利用することができます。例えば、%.0f は整数化して表示し、%.6g は有効桁数を 6 桁とするフォーマットです。



## 第 11 章

# ゲーム


本章には Vensim PLE は対応しません。

### 11.1 ゲームとは？

ゲームは、シミュレーションが進む過程に積極的に関与する手法です。ゲームは、フライトシミュレータのようなもので、利用者がその都度、シミュレーション結果に影響する意思決定を行います。Vensim のシミュレーションモデルは、ゲームとして実行することができます。その際、時間ステップごとにゲーム変数を変更することができます。これに対して、通常の Vensim のシミュレーションモデルは、初期設定で与えられたシミュレーションの時間の範囲で実行されます。

フライトシミュレーションは、航空機などの飛行の操縦を模擬するもので、操縦装置の操作に対する機体の反応をコンピュータで計算し、結果を操作パネルで表示するようになっています。

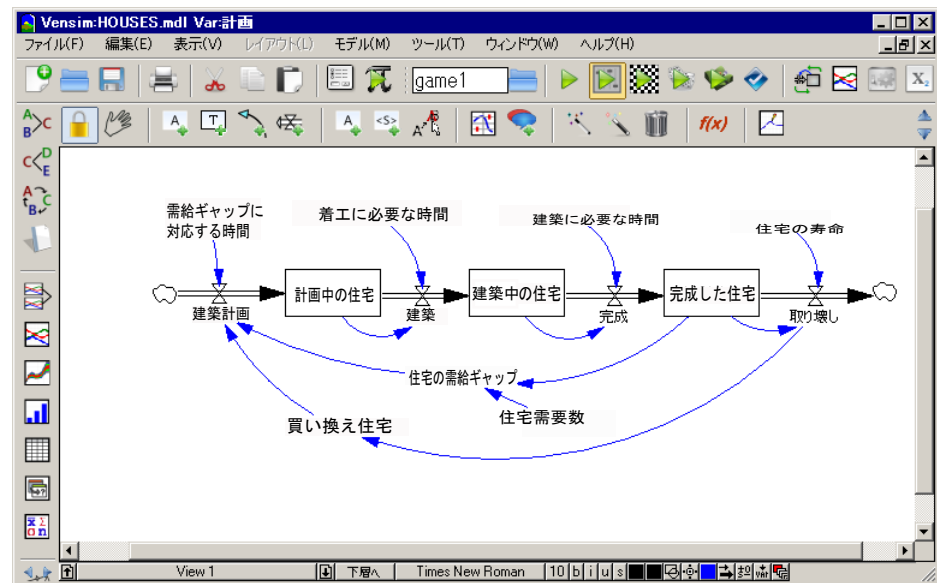
#### 11.1.1 不動産ゲーム (houses.mdl)

- 開く  をクリックし、*guide\chap11* から、*houses.mdl* を開きます。もしくは、
- 以下の図や方程式を参考にモデルを作成し、*guide\chap11* に、異なる名前（例えば *myhouses.mdl*）を付けて保存します。[ 時間の範囲 ] は、[ 開始時間 ] を「0」、[ 終了時間 ] を「100」、[ 時間ステップ ] を「0.5」とし、[ 時間単位 ] を「月」と設定します。

英語版です。

#### モデルの構造

これは不動産業における建設モデルです。家の需要ギャップの解消と家の完成には長い遅れがあります。このモデルの特徴はいくつかの遅れを持つ負のフィードバックループにあります。



houses.mdlの方程式

住宅の寿命 = 1200

Units : 月

建築 = 計画中の住宅 / 着工に必要な時間

Units : 住宅/月

完成 = 建築中の住宅 / 建築に必要な時間

Units : 住宅/月

取り壊し = 完成した住宅 / 住宅の寿命

Units : 住宅/月

住宅の需給ギャップ = 住宅需要数 - 完成した住宅

Units : 住宅

完成した住宅 = INTEG (完成 - 取り壊し, 5000)

Units : 住宅

計画中の住宅 = INTEG (建築 - 完成, 建築 \* 建築に必要な時間)

Units : 住宅

建築計画 = MAX ( 0, 買い換え住宅 + (住宅の需給ギャップ / 需給ギャップに対応する時間) )

Units : 住宅/月

買い換え住宅 = 取り壊し

Units : 住宅/月

建築に必要な時間 = 6

Units : 月

着工に必要な時間 = 3

Units : 月

需給ギャップに対応する時間 = 8

Units : 月

### 組み込み関数

このモデルを、均衡状態を持つモデルとしてまず作成します。「住宅の需要」を5000に設定します。これは「完成した住宅」の初期値です。したがって「住宅の需給ギャップ」が0となります。つまり「建築計画」は「住宅の買い換え」や「取り壊し」と等しくなります。またこれら以外のストック変数についても、均衡が成立するように設定します。例えば「計画中の住宅」の初期値は、「建築計画」\*「着工に必要な時間」となっています。このように、シミュレーションを実行しても、値が一定であるシミュレーションモデルになっています。

モデルが正しく作成されていること、また実行結果が均衡を保つことを確認することは重要です。ここではさらに、モデルの振る舞いを詳しく見ます。「住宅の需要数」を5000の定数とするのではなく、10ヶ月の間だけ5000とし、その後5050に増大させます。そのために、次のような方程式とします。

**住宅の需要数** = 5000 + STEP ( 50, 10)  
Units :住宅

STEP 関数には、[height]【高さ】と [start time]【開始時間】という2つの引数があります。それらは、括弧{}で閉じられています。[start time] に到達するまでは、0の値を返し、それ以降は [height] を返します。この関数はモデルの入力に大変適しています。なぜなら、簡単な入力変更で広範な振る舞いを起こすためです。このようなモデルに“刺激”や“攪乱”を与える関数には、STEP 以外に PLUSE や RANMP があります。

上記の方程式を追加するために、方程式  をクリックし、「住宅の需要数」の方程式編集ダイアログを開きます。

- 編集ボックスに「5000」と入力し、「+」を入力します。
- [関数] タブをクリックし、関数リストから [STEP] を選択します。[Add sel]【選択された関数を追加】をクリックします。
- 引数{height}が強調表示されているので、「50」を入力します。
- 引数{stime}をダブルクリックし、「10」を入力します。
- [単位] を「住宅」とし、[OK] をクリックします。

### WIP グラフ

- 以下のようにしてカスタムグラフを定義します。特に、[As WIP Graph(maxpoints)] **WIP グラフは進行と共に変化するグラフです。**  
【WIP グラフの作成】にチェックが入っていることを確認してください。10 章ではカスタムグラフの作成方法を説明しています。

変数	タイプ	単位	ラベル	LineW	単位	Y-min	Y-max
完成した家	選択			2		5000	5150
計画	選択			2		0	20
家の需要ギャップ	選択			3		-100	100

### ゲーム変数の追加

このゲームの目標は、住宅の需給を満たすこと（「住宅の需給ギャップ」をゼロにすること）です。そのために変数「建築計画」の値を設定し、「計画中の住宅」や「建築中の住宅」を制御します。今のところ「建築計画」には式が与えられています。この式によってシミュレーションを実行することができますが、シミュレーションの途中に「建築計画」の値を変更する手段が備わっていません。そのためには、変数「建築計画」をゲーム変数に変更します。

- 方程式  $f(x)$  を選択します。
- 「計画」をクリックします。

以下の方程式が書かれています。

$$\text{計画} = \text{MAX}(0, \text{買い換え住宅} + (\text{住宅の需給ギャップ} / \text{需給ギャップに対応する時間}))$$

Units 住宅/月

この方程式によって、「建築計画」はマイナスにはなりません。数件の家の建設計画すること（正）、もしくは、建設計画を持たないこと（ゼロ）場合があります。この「建築計画」をゲーム変数にするために、変数のタイプを変更します。

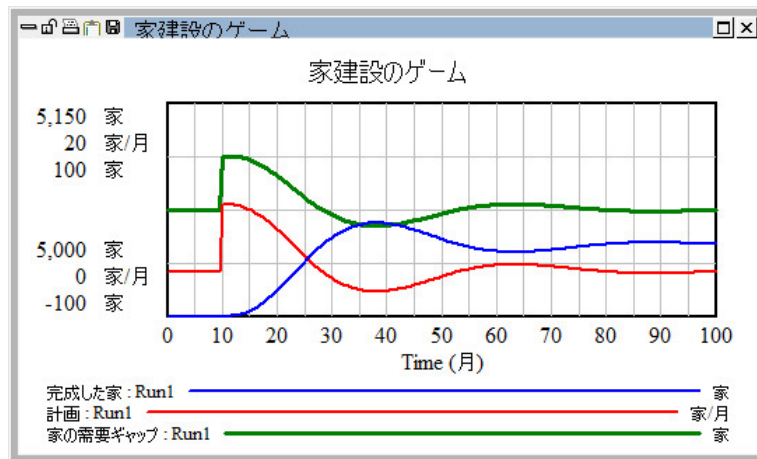
- 方程式編集ダイアログの[タイプ]の下段のドロップダウンボックス[▼]（[標準] 1）になっています）をクリックし、リストから[Gaming]【ゲーム】を選びます。
- [OK] をクリックします。
- モデルを保存します。

このようにして、補助変数やフロー及び定数をゲーム変数に変更できます。シミュレーションの実行中では、ゲーム変数は元の補助変数やフロー及び定数のように働きます。しかしながら、ゲームの実行中では、その進行に合わせて、ゲーム変数の値を設定することができます。

### モデルのシミュレーション

ゲームを始める前に、シミュレーションを実行した時のモデルの振る舞いを見えます。

- シミュレーション名の編集ボックス  をダブルクリックし、「Run1」と入力します。シミュレーションの実行  をクリックします。






A Work-In-Progress【進行とともに変化する】(WIP) カスタムグラフが作成され、「完成した住宅」「建築計画」「住宅の需給ギャップ」の3つの変数の振る舞いが表示されます。その振幅、つまり目標を上回ったり、下回ったりする動きに注目します。このモデルは「住宅の需給ギャップ」をゼロにしようとしています。モデルの変化は、「住宅の需給」の急増から生じています。「住宅の需給ギャップ」をよりゼロに近づけるための、より良い建設計画をゲームを用いて考えます。

### 11.1.2 ゲームの実行

- シミュレーション名の編集ボックス  をダブルクリックし、「game1」と入力します。ゲームの実行  をクリックします。

A WIP (Work-In-Progress) カスタムグラフが作成され、ツールバーがゲームツールバーに変わります。





ゲームの時間が左側に表示されます。ゲームの実行中にゲーム変数の値を変更する方法として、ゲーム変数の変更  $\pi$  があります。停止  はゲームを停止します。前進  と後退  は、増減時間編集ボックス (0.5 と表示されています) に示した値だけゲームを進めます。


### 11.1.3 ゲームの前進

- スケッチが隠れないように、WIP グラフを右に移動します。


変数「建築計画」がピンク色の背景により強調表示されます。これは、ゲームの実行中にゲーム変数を変更する2つ目の方法です。

- 「建築計画」をクリックします。初期値が (4.166) となっています。値を変更せずにそのまま  を押します。




WIP グラフがモデルウィンドウの背後に隠れます。ツールバーにあるゲーム変数の変更  によってゲーム変数を変更した場合、WIP グラフはそのままの状態、一番手前に表示されます。2つのウィンドウをどちらも表示させるには、両方のサイズを小さくし、画面に収まるようにします。

- モデルウィンドウの元に戻す (縮小)  をクリックします。これは右上にありますが、Vensim の元に戻す (縮小) ボタンの下に配置されています。



スケッチを含むモデルウィンドウが小さくなります。

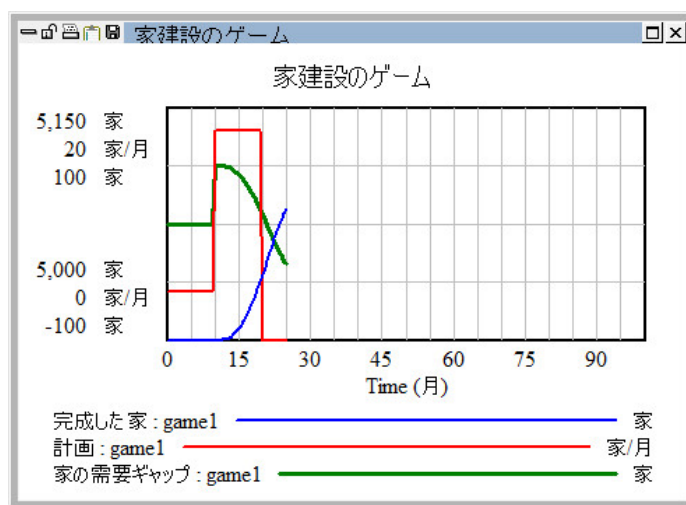
- モデルウィンドウとグラフウィンドウのサイズと位置を調整することで、モデル (少なくとも変数「計画」) と WIP グラフを見ることができます。
- 増減時間編集ボックスをダブルクリックし、「5」を入力します。
- 前進  をクリックします。

WIP グラフが表示されます。均衡が保たれています。「住宅の需給ギャップ」がゼロなので、どの値も変更する必要はありません。

- 前進  をもう一度クリックすると、「住宅の需給ギャップ」がステップアップします。
- 「建築計画」をクリックし、「18」を入力します。  を押します。
- 前進  を 2 回クリックします。


「完成した住宅」が上昇するなかで、「住宅の需給ギャップ」が縮小しています。住宅の需給ギャップはゼロに近づきました。家を大量に建設することは止めた方がよい。

- 「建築計画」をクリックし、「0」を入力します。  を押します。
- 前進  をクリックします。






なんということでしょうか。基準を超えてしまいました。「住宅の需給ギャップ」はマイナス（中央の線より下）になります。家をマイナス建設することはできないので、しばらくゼロを続けます。




- 「住宅の需給ギャップ」が正の値を示すまで（Time=50 でちょうどゼロになります）、前進  をクリックします。

「住宅の需給ギャップ」が正の値を示さないように（家の需要が増加するので）、再度住宅の建設を始めます。こうした事態の発生を予測することができるので、「住宅の需給ギャップ」が正の値を示す前に家を建設することもできます。

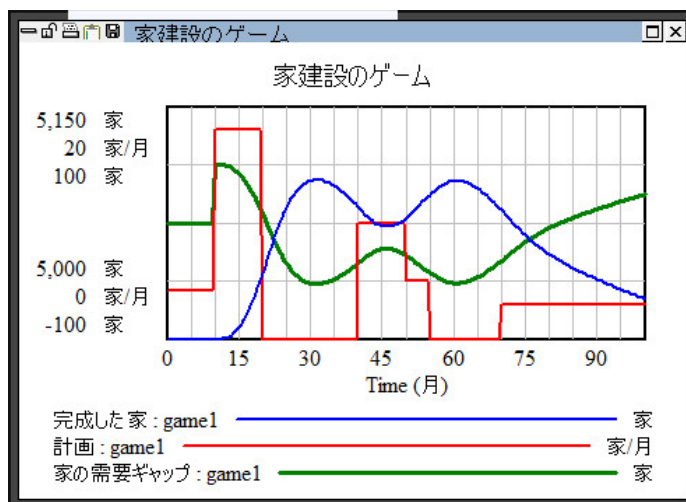
#### 11.1.4 ゲームのさかのぼり

- 後退  を2度クリックします。「住宅の需給ギャップ」が負の値を示します。

もちろん、現実ではさかのぼりはありません。しかし、ゲームが思わぬ結果をもたらした場合、別の選択肢を試すためにゲームのさかのぼりが可能です。

- 「建築計画」をクリックし、大きめの値として、「10」を入力します。 を押します。
- 前進  をクリックします。
- 「住宅の需給ギャップ」を、ゼロか、ゼロに近い状態で維持するようにゲームを続けます。ゲームは終了時間の「100」まで行います。
- 停止  をクリックします。

WIP グラフは、以下のようなものになります。



このグラフと同じ結果を得るには、  
 0 から 9.5 までは初期値、  
 10 から 19.5 までは 18、  
 20 から 39.5 までは 0、  
 40 から 49.5 までは 10、  
 50 から 54.5 までは 5、  
 55 から 69.5 までは 0、  
 70 からは 3、とします。

ゲームの結果は、おそらく元のシミュレーション結果に比べてあまり良いものではありません。場合によっては、悪いことも考えられます。このグラフでは、ゼロを維持しようと努めた「住宅の需給ギャップ」は、家建設の「建築計画」によって大きく変動します。



## 第 12 章

# 入出力コントロール

モデル表示画面に入力と出力のコントロールを追加し、シミュレーションモデルでのスケッチをカスタマイズすることができます。入出力コントロールは、モデル構造と一緒に、または別の表示画面に追加することができます。入出力コントロールの使用によって、モデルの入力を管理したり、シミュレーション結果を見るための、独自の“コントロールルーム”を作成できます。

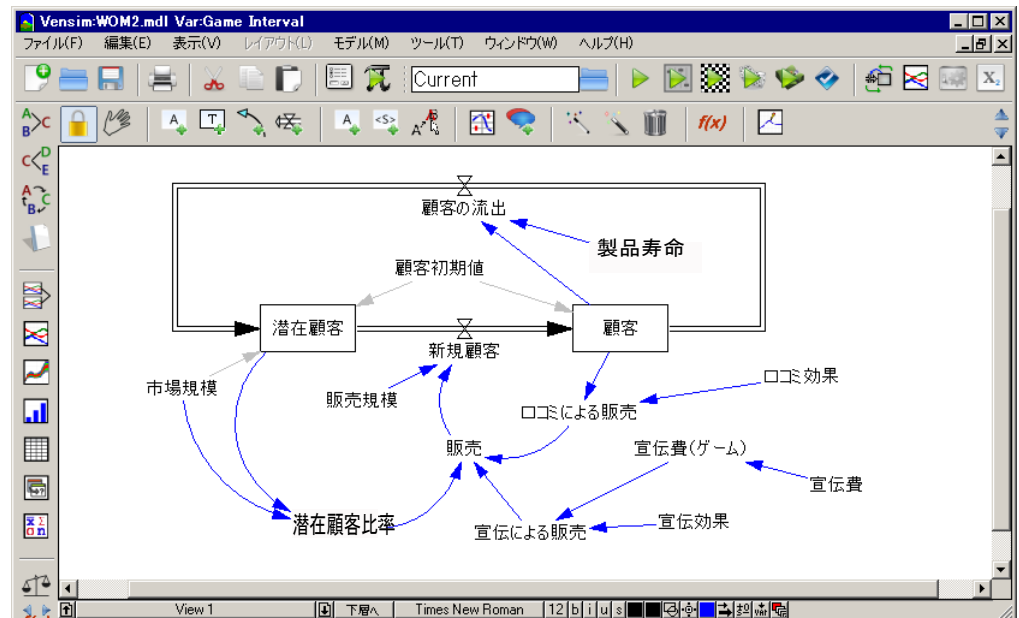
本章では、既存のモデルのコントロールルームを作成します。入出力コントロールは、モデル構造の一部ではありません。そのため、モデルのシミュレーションには影響を与えません。それらは、他の人がモデルを容易に使用するために、完成したモデルに追加されたものです。入出力コントロールは、Vensim Model Reader でサポートされており、モデルを広く利用するための機能を提供します。入出力コントロールは、モデル構造にも対応しています。ある変数の名前を変更した場合、対応するコントロールが自動的にその名前を更新します。しかし、もし変数を削除したり、種類を変更したりすると、コントロールは対応しません。しかし、これはモデルの操作を止めるものではなく、コントロールによる出力が表示されないだけです。

### 12.1 口こみによる販売数

ここでは、「口こみによる販売数」のモデルを使います。このモデルの作成方法については、モデリングガイドの 4 章で扱っています。

- *guide\chap12* から、*wom1.mdl* を開きます。  
または、
- 以下のような図と方程式のモデルを作成し、*guide\chap12* に、*wom2.mdl* として保存します。[ 時間の範囲 ] は、[ 開始時間 ] = 「0」、[ 終了時間 ] = 「5」、[ 時間ステップ ] = 「0.0625」とし、[ 時間単位 ] は「年」とします。

## 12.1.1 wom1.mdl の方程式



宣伝効果 = 0.1  
Units : 個/円

宣伝費 (ゲーム) = GAME (宣伝費)  
Units : 円/年

宣伝費 = 2e+007  
Units : 円/年

口コミ効果 = 3  
Units : 個/人/年

顧客 = INTEG (新規顧客 - 顧客の流出, 顧客初期値)  
Units : 人

潜在顧客比率 = 潜在顧客 / 市場規模  
Units : Dmnl

顧客の流出 = 顧客 / 製品寿命  
Units : 人/年

市場規模 = 1e+008  
Units : 人

新規顧客 = 販売 / 販売規模  
Units : 人/年

製品寿命 = 2  
Units : 年

潜在顧客 = INTEG (顧客の流出 - 新規顧客, 市場規模 - 顧客初期値)  
Units : 人

販売 = (口コミによる販売 + 宣伝による販売) \* 潜在顧客比率  
Units : 個/年

販売規模 = 1  
Units : 個/人

顧客初期値 = 10000  
Units : 人

宣伝による販売 = 宣伝費 (ゲーム) \* 宣伝効果  
Units : 個/年


口こみによる販売 = 顧客 \* 口こみ効果  
Units : 個/年

## 12.2 出力コントロール

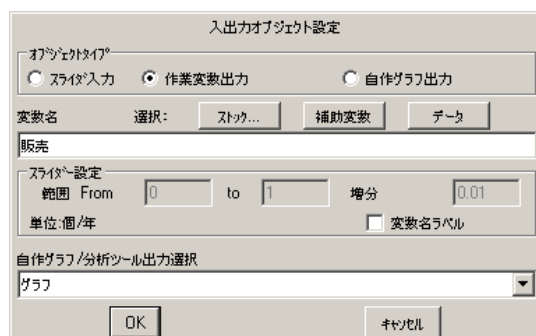
出力コントロールに加える新しい表示画面を作成します。

- ステータスバーにある表示画面  View 1  をクリックし、[ \*\*New\*\* ] をクリックします。

新しい空白の表示画面が開きます。

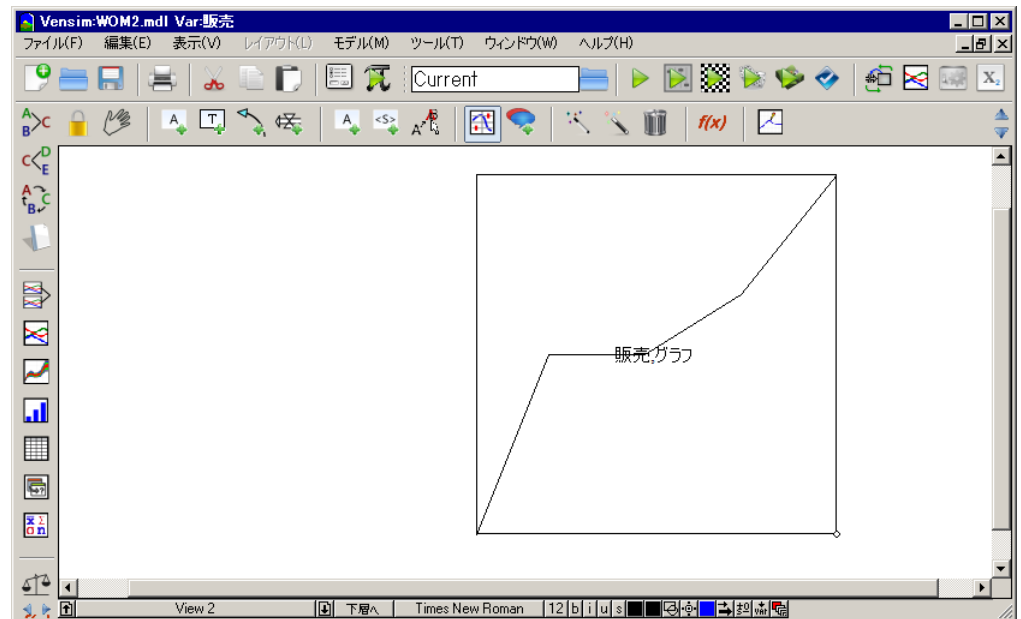
- 入出力オブジェクト  をクリックするか、**[0]** を押します。
- カーソルをスケッチの右方向に移動させ、クリックします。入出力オブジェクトの設定ダイアログが開きます。
- [ オブジェクトタイプ ] の [ 作業変数出力 ] を選択します。[ 補助変数 ] をクリックし、変数のリストの中から [ 販売 ] を選びます。[ OK ] をクリックし、変数選択のダイアログを閉じます。
- [ 自作グラフ/分析ツール出力選択 ] のドロップダウンボックス[ ] をクリックし、[ Graph ] を選択します。


ダイアログは以下のようになります。



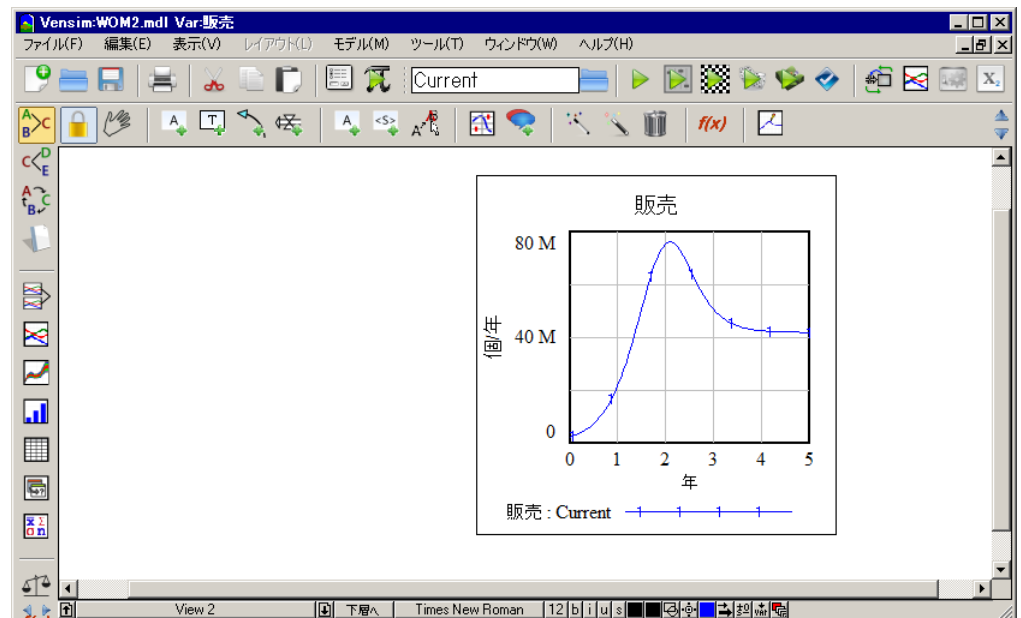
- [ OK ] をクリックします。

サイズ調整のハンドルを持つ、かなり大きな長方形が現れます。スケッチの右側に収まるようにサイズを変更します。以下ようになります。



➤ シミュレーションの実行  をクリックします。(または、**Ctrl**+**R**を押します。)


モデルがシミュレーションされ、グラフが以下のように表示されます。



### 12.3 入力コントロール

同じ表示画面で操作を続けます。

### 12.3.1 宣伝費のスライダ

- 入出力オブジェクト  を選択し、スケッチの左上をクリックします。入出力オブジェクトの設定ダイアログが開きます。
- [オブジェクトタイプ]の[スライダ入力]を選択します（おそらくデフォルトで選択されています）。
- [定数]をクリックし、リストから、[宣伝費]を選択します。
- [スライダー設定]の[範囲]の[From]に「0」、[To]に「200e8」、そして[増分]に「0.5e6」と入力します。
- [変数名ラベル]のチェックをはずします。


200e8 は 20 億円、0.5e6 は 5 千万円

入出力オブジェクトの設定ダイアログは、以下のようになります。



- [OK] をクリックします。


スライダが表示されます。

- グラフに合わせて、サイズを調節します。
- スケッチコメント  を選択し、スライダの上をクリックします。
- コメント欄に、「宣伝費（円/年間）」と入力します。[形状]の[空ボックス]を選択し、[OK] をクリックします。これは、作成したスライダのラベルです。


### 12.3.2 製品寿命のスライダ

「老朽化時間」についても、同じようにスライダを作成します。[範囲]を「0.5」から「10」、[増分]を「0.1」とします。スライダの上には、「製品寿命（年間）」というラベルを付けます。

### 12.3.3 他の変更可能な定数

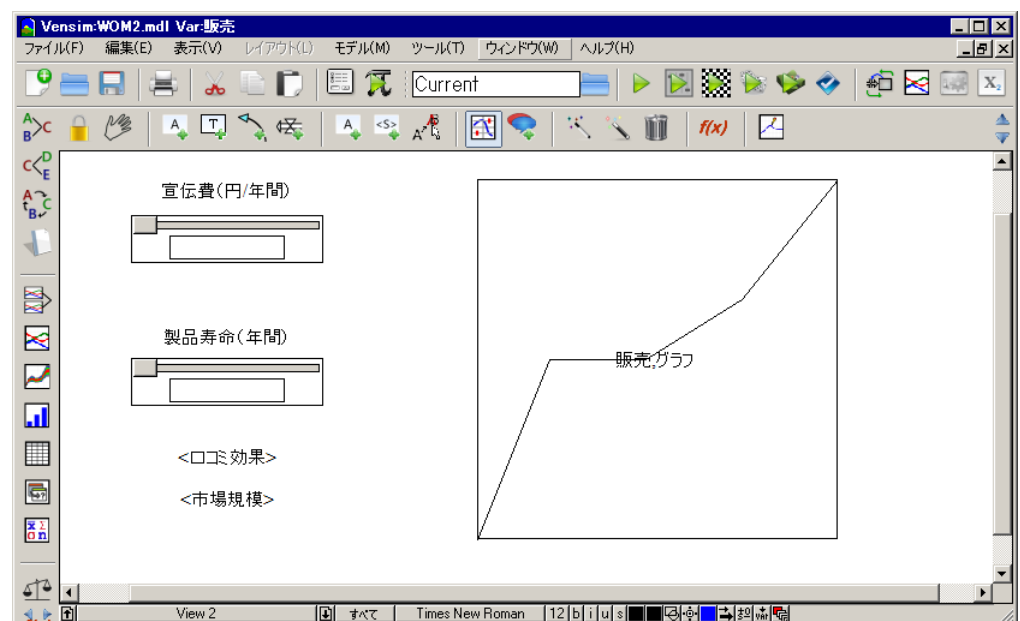
- 代行変数  を選択し、定数「口こみ効果」と「販売規模」を追加します。
- 新しい代行変数を選択（強調表示）し、ステータスバーのボタンを使って、文字の色を黒にします。または、この代行変数を右クリックし、ダイアログの中で、文字の色を黒にします。

### 12.3.4 整列

- 追加したものをきれいに並べるために、スケッチ移動  を選択し、左側のすべての要素が含まれるようにドラッグし、強調表示します。
- **[Shift]**を押しながら、一番上の「宣伝費（円/年間）」をクリックし、強調表示を解除します。そして、もう一度クリックして、再度強調表示します。
- **[レイアウト]**メニュー **[中央に揃える]**を選びます。
- 選択された全ての要素を、適当な位置にドラッグします。


これで完成です。

要素の整列は、一番最後に選択された変数を基準にします。ここでは、「宣伝費（円/年間）」が基準になります。




新しい名前で、モデルを保存します。*wom1.mdl* に追加された変更は、*guide\chap12*にある *wom2.mdl* に保存されています。ここで作成したものと比較することができます。


### 12.3.5 モデルのシミュレーション

- シミュレーションの準備  をクリックします。(または、**[Ctrl]+[E]**を押します。) 新しいシミュレーション名を入力します。

スライダが稼働し、2つの代行変数が強調表示されます。

- スライダをドラッグし、シミュレーションの実行  をクリックします。(または、**[Ctrl]+[R]**を押します。)

モデルがシミュレーションされ、新しい結果がグラフに表示されます。これを繰り返して、異なる変化も見ます。シミュレーション名を変えることもできます。シミュレーションによる結果は、その都度右側のグラフに表示されます。


また、統合シミュレーションの実行  をクリックし、スライダを動かしながら、自



動的にグラフが更新されるのを見ることもできます。この場合、「口こみ効果」と「販売規模」にもスライダが表示されます。



## 12.4 ゲームコントロール

「口こみによる販売数」のモデルは、「宣伝効果」というゲーム変数を持っています。シミュレーションコントロールの設定と同じように、ゲームコントロールでも同様のレイアウトを使用します。




- スケッチ移動  を選択します。
- 右側の出力オブジェクト及び、コメント「宣伝費（円/年間）」と、その下のスライダを強調表示させます。

同じことは、どれか1つをクリックしたあと、**[Shift]**を押しながら、他のものをクリックしても行えます。

- **[編集]メニュー** **[コピー]**を選びます（または、**[Ctrl]+[C]**を押します）。

**注意** スケッチ移動  を使って入出力オブジェクトをコピーするのが、もっとも容易な方法です。スケッチ固定  が選択されている時、これらのオブジェクトは、クリックや**[Shift]**を押しながらのクリックでは選択できません。しかし、マウスをドラッグして選択することはできます。

### 12.4.1 新しい表示画面への貼り付け



- 新しい表示画面を作成するため、ステータスバーにある表示画面  **View 1**  をクリックし、**[\*\*New\*\*]**をクリックします。
- **[編集]メニュー** **[貼り付け]**を選びます（または、**[Ctrl]+[V]**を押します）。
- 貼り付けのダイアログの**[OK]**をクリックします。貼り付けられる変数は構造を持たないため、Replicate【複製】とPaste Picture【貼り付け】は同じ結果になります。
- 入出力オブジェクト  を選択し、スライダをクリックします。

入出力オブジェクトのダイアログが開きます。

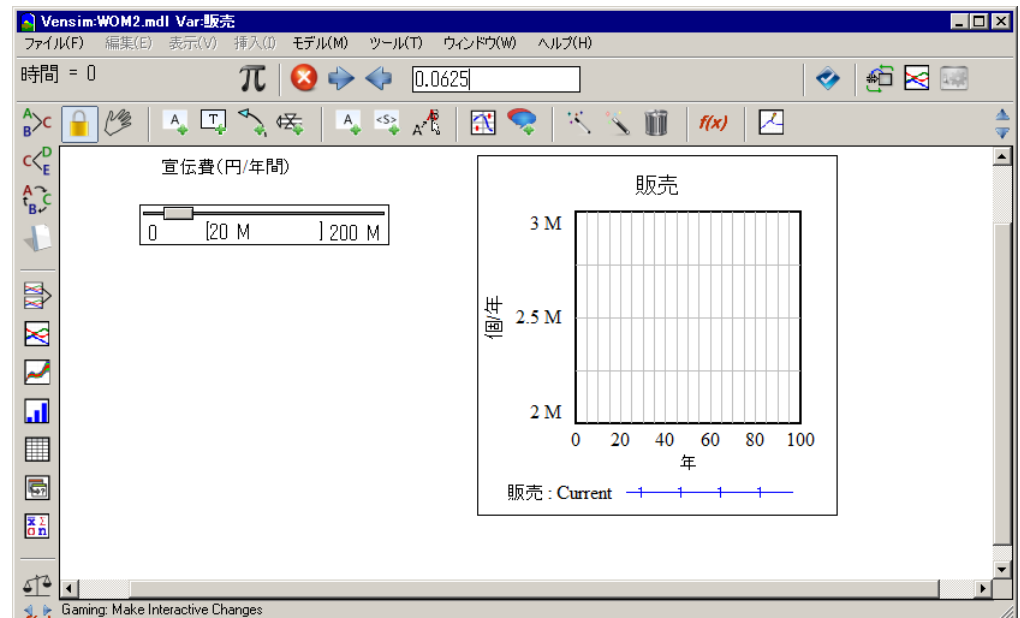
- **[ゲーミング]**をクリックします。**[宣伝効果（ゲーム）]**を選択します（リストにあるのは**[宣伝効果（ゲーム）]**のみです）。**[OK]**をクリックします。

スライダによりコントロールされる変数は「宣伝費（円/年間）」のままなので、ラベルを変更する必要はありません。


### 12.4.2 ゲームの実行


- 制御パネル  をクリックし、**[データセット]**タブで、すべてのデータセットを読み込まないようにします。
- ゲームの実行  をクリックし、ゲームを始めます。

グラフが表示されます。おそらく時間の目盛り軸が 0 から 100 のグラフが表示されます。この時間の目盛りは、スライダの動きに合わせて調整されます。スライダが稼働し、ツールバーがゲーム仕様になります。



ゲームの間隔がツールバーの編集ボックスに強調表示されます。ゲームの間隔の初期設定は [ 時間ステップ ] と同じに設定されており、この場合「0.0625」です。

- 「0.25」と入力します。
- スライダを使って「宣伝費」を変化させ、前進  をクリックして、時間を進めます。

グラフが更新されます。後退  を使って、時間をさかのぼることもできます。

モデルを保存します。追加された変更は、*guide\chap12\Complete* にある *wom3.mdl* に保存されています。ここで作成したものと比較することもできます。

## 12.5 モデルの公表

作成したモデルは、定数を変更したりシミュレーションを実行したりするための、簡単なインターフェイスを持っています。また、ゲームを実行するための簡単なインターフェイスを持ちます。ここで紹介するナビゲーションボタンは、表示画面の操作について知らない人にとって、役に立ちます。さらに、設定された [ 時間ステップ ] よりも、適切なゲームの間隔を組み込むことができます。

### 12.5.1 ゲームの間隔

ゲームの間隔を、より妥当な値にするには、「Game Interval」という定数をモデルに追加します。これは補足的な変数とし、その値は「0.5」年とします。ゲームを実行すると、定数 0.5 がゲームの間隔として読み込まれます。

### 12.5.2 コメントとナビゲーション

モデルの初心者のために、モデルを知るための説明やガイドを準備する必要があります。

- ステータスバーにある表示画面  View 1  をクリックし、  
[ \*\*New\*\* ] をクリックします。

#### 表示画面の改名

- [ 表示 ] メニュー [ 名前変更 ] を選択します。ダイアログが開いたら、「概観」と入力し、[ OK ] をクリックします。

これで、表示画面の名前が「概観」になりました。同じように、他の表示画面も「構造」、「シミュレーション」、「ゲーム」に変更します。

#### 表示画面の再配列

- [ 表示 ] メニュー [ 再配列 ] を選択します。
- 表示画面のリストの中から、「概観」をクリックします。
- マウスを押したまま、上の方へドラッグします。





「概観」が隠れ、ポインタの形が、十字に変わります。


- 十字を、一番始めの名前「構造」の上に移動します。マウスボタンを離します。

「概観」が一番上に移動します。もしうまくいかない場合は、マウスを離す位置を少し下げて、繰り返します。もし、十字の中央がリストの外側に移動してしまった場合、「概観」は元の場所に戻ります。


- [ OK ] をクリックします。


#### コメントの追加

もし表示画面が [ 概観 ] になっていなければ、表示画面  View 1  をクリックするか、 と  を使って、[ 概観 ] を選択してください。

- スケッチコメント  を選択し、画面の上の方をクリックします。
- 役立つと思うコメントを、いくつか入力します。以下の追加したコメントを見えます。

### 12.5.3 ナビゲーション

ナビゲーションは、行き先を持つコメントです。スケッチ固定  を選択してナビゲーションをクリックすると、自動的に行き先に指定された表示画面に変わります。

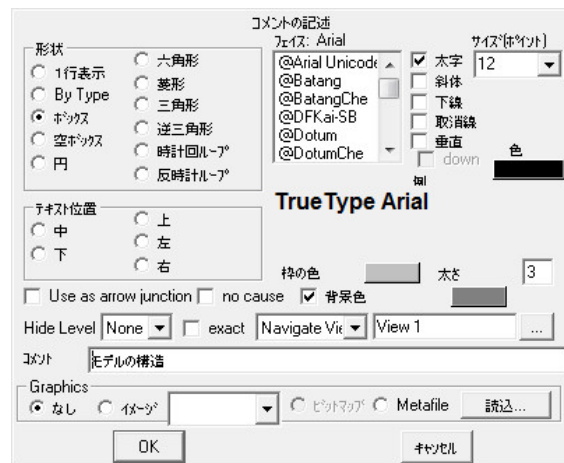
- スケッチコメント  が選択されたままで、先ほどのコメントの下をクリックします。

- 「モデルの構造」と入力し、[ 形状 ] で [ ボックス ] を選択します。
- [ Navigate View ] の右側にある [ ... ] をクリックします。

新しいダイアログが開き、表示画面のリストが現れます。

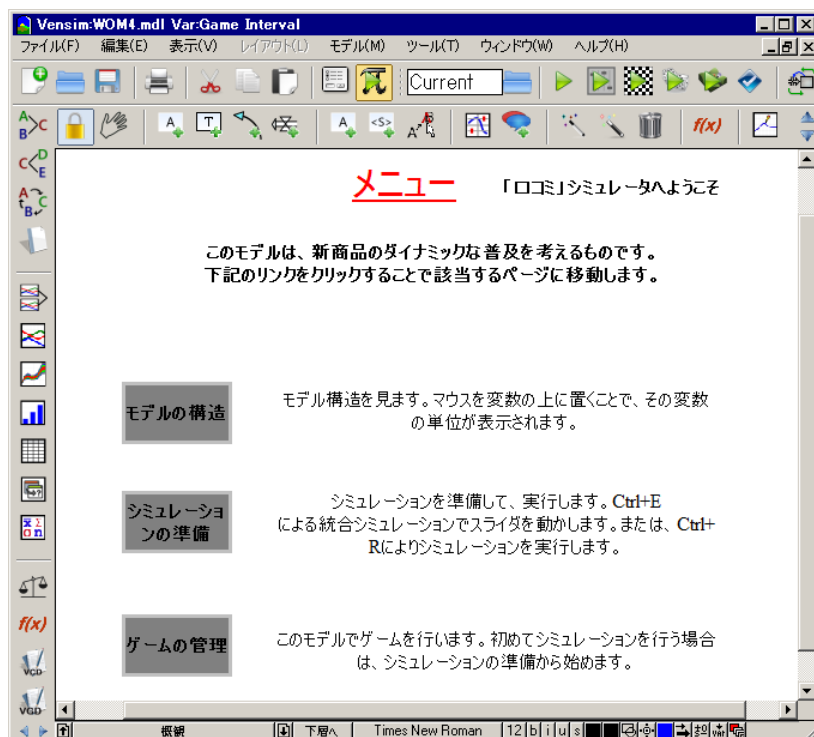
- リストから [ 構造 ] をえらび、[ OK ] をクリックします。
- [ 枠の色 ] の右側のボタンをクリックし、濃いグレーをクリックします。
- [ 背景色 ] の右側のボタンをクリックし、淡いグレーをクリックします。
- [ 太さ ] に 2 を入力します。

背景が淡いグレーで、枠が濃いグレーでできた、ボタンのようなコメントができます。  
コメントダイアログは以下のようになります。



- 他の表示画面にも、それぞれ「シミュレーションの準備」と「ゲームの管理」というラベルで、ナビゲーションを設定します。



表示画面「概観」は以下のようになります。



- 他の表示画面から「概観」に戻るためのナビゲーションを追加します。また、「シミュレーション」に、「ゲーム」へのナビゲーションを追加します。

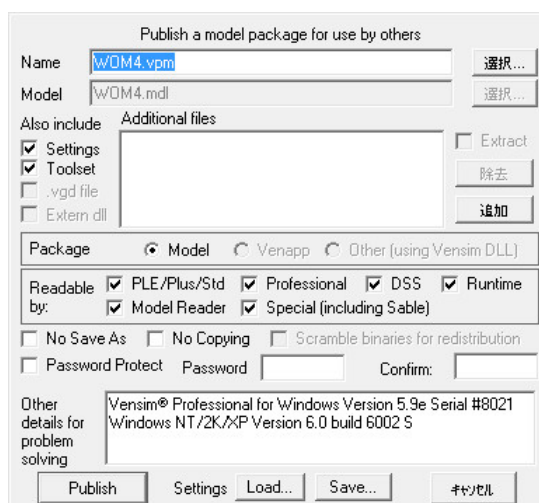
モデルを保存します。追加された変更は、*guide\chap12\Complete* にある *wom4.vmf* に保存されています。ここで作成したものと比較することもできます。通常の拡張子である *.mdl* ではなく、*.vmf* が使われます。その理由は次節で示します。

#### 12.5.4 稼働テスト

スケッチ固定  を選択し、ナビゲーションを試してみます。また、シミュレーションの準備と実行を繰り返します。シミュレーションの準備において、表示画面「ゲーム」を再検討してください。ゲームにおいて、表示画面「シミュレーション」を再検討してください。スケッチ固定  を選択した状態での操作は、基本的に、Vensim Model Reader での操作と同じです。

#### 12.5.5 パッケージの公表

Vensim Model Reader は、テキスト形式 (*.mdl*) のモデルを読むことのできない、読み込み専用プログラムです。そのため、Vensim Model Reader を使用する前に、特別のバイナリ形式で作成したモデルを保存する必要があります。その方法は、[ファイル]メニュー [出版] を選びます。



もしシミュレーションを実行したことがあれば、ダイアログに実行ファイルのリストが表示されます。このモデルではシミュレーションの実行結果は必要ないので、それぞれのリストの名前をクリックし、[ 除去 ] をクリックします。

また、[ ファイル ] メニュー [ 別名で保存 ] を選び、モデルをバイナリ形式 ( *.vmf* ) で保存することもできます。ダイアログの [ ファイルの種類 ] で、[ Binary Format Models ] を選択し、名前を入力して保存します。名前に拡張子 *.vmf* を付けることもできます ( 例えば *wom4.vmf* )。Vensim は入力した拡張子から、保存するファイルの種類を決定します。

出版によるパッケージ化は、モデルを開いたり、その利用方法を管理することができます。これは、19 章でより詳しく説明しています。バイナリ形式での保存は、テキスト形式と同様に開いたり、修正したりすることができます。バイナリ形式のファイルは、ファイルサイズが大きく、また Vensim からしか利用できないため、記録するには適していません。しかし、バイナリファイルは、リファレンスマニュアルの 8 章にあるように、シミュレーションコントロールダイアログ上で容易に動くという利点があります。また、バイナリ形式のモデルは、Vensim DSS リファレンスで紹介される、コンパイルされたシミュレーションで利用されます。

フリーの Vensim Model Reader をダウンロードすることで、誰にでも利用できるモデルを作成できます。

## 第 13 章

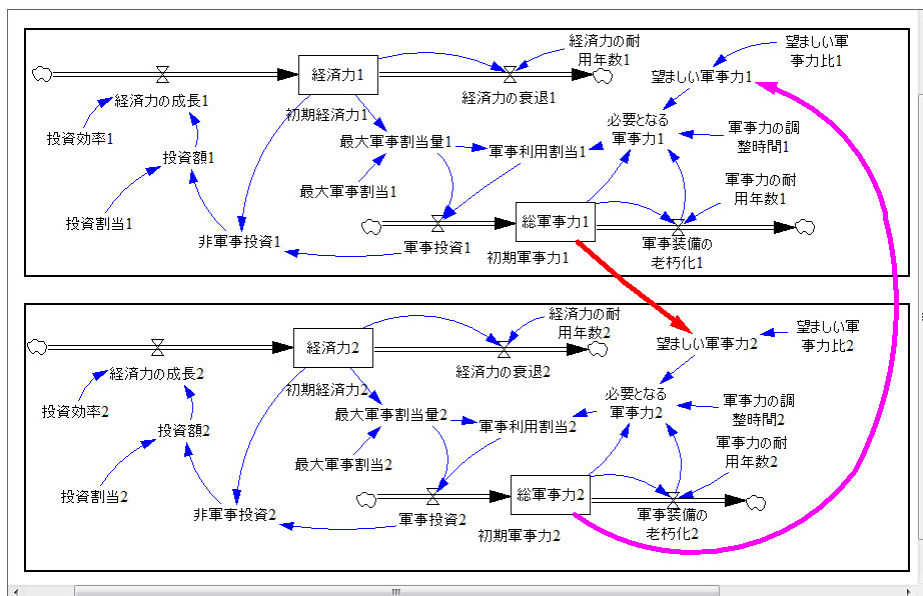
# 統合シミュレーション

従来シミュレーションモデルはバッチ処理で行われてきました。つまり、いくつかの仮定を設定し、シミュレーションし、その結果を見るという手続きです。25 年前には、これが必要不可欠なものでした。コンピュータは、大型で高価なものであり、一方で処理速度は遅いものでした。

技術は大きく進歩しましたが、シミュレーションのバッチ処理は、これまでと同様のままでした。ソフトウェアの進歩の過程で、シミュレーションの結果を表示することが可能になりましたが、実際には、シミュレーションにおける計算や再検討には変化をもたらしていません。Vensim のバージョン 5 では、モデルにおける計算や再検討の関係を同時に分析可能となり、驚くほど大きなモデルでも、それを利用できます。このアプローチを統合シミュレーションといいます。

### 13.1 軍備競争モデル

以下のモデルを作成するか、`guide\chap13\complete\arms2.mdl` を開きます。これは 英語版です。経済成長を含んだ軍備競争モデルです。





英語版です。

### 13.1.1 図の新しい装飾


上の図は、2 つの点を除いて、簡単なモデルです。第 1 は、2 国の軍備競争を囲む大きな長方形です。第 2 は、2 国をつないでいる、長く、循環していない矢印による接続です。これから紹介するのは、これらの作成の方法です。guide\chap13\complete\arms1.mdl から始めることもできます。

### 13.1.2 長い接続


「総軍事力 2」から「望ましい軍事力 1」に延びる矢印は通常の矢印ではありません。これは、矢印と結ぶことができるコメント（これを junctions【結合】といいます）への、小さな矢印で作成されています。これを作成するには、まず結合を作成し、その後矢印を追加します。そして、不要な矢頭を削除し、結合のサイズを 0 とします。

- スケッチコメント  を選択します。
- 「軍事装備の老朽化 1」の右側の空白をクリックします。
- 「背景色」を青にします。
- [ Use as arrow junction ]【矢印の結合に利用】にチェックを入れます。
- [ OK ] をクリックします。これを結合箱と呼びます。
- 結合箱を小さくします。
- 「軍事装備の老朽化 2」の右側にも、同じように結合箱を作成するか、またはコピーして貼り付けます。
- 矢印  を選択します。
- 「総軍事力 2」から、下の結合箱に矢印を描きます。
- 下の結合箱から、上の結合箱に矢印を描きます。
- 上の結合箱から、「望ましい軍事力 1」に矢印を描きます。
- 下の 2 つの矢印を、それぞれ右クリックし、[ 矢頭 ] のチェックをはずします。  
[ Line Style/Thickness ]【線の形状と幅】において 3 番目に太い幅を選択し、[ 色 ] をピンクとします。
- 上の矢印の幅と色も、同じように設定します。
- 結合箱と矢印の位置を調整します。
- 結合箱のサイズを 0 とします。右下のハンドルを押して、見えなくなるまで左上に動かします。

### ヒント

サイズを 0 にした後、結合箱を移動させたい場合は、スケッチ移動  を選択し、結合箱のまわりを取り囲むようにドラッグし、選択された長方形を移動させます。

### 13.1.3 収納枠


- スケッチコメント  を選択し、スケッチの左上の空白部分をクリックします。
- ダイアログの[ 形状 ]の[ ボックス ]を選択し、[ 太さ ]を「2」と入力します。[ OK ]





をクリックします。

- 上半分のモデルを取り囲むように、サイズを調整します。

#### 注意

枠を作成した後、サイズを調整するハンドルが数秒間表示されます。その後ハンドルが消えても、スケッチ移動  をクリックすれば、再度表示されます。

- スケッチ固定  をクリックします。
- 新しく作成した枠をクリックし、強調表示します。
- ステータスバーの、ハイライトされた変数の順位を最背面に移動する  をクリックします。強調表示が消えます。
- 下半分のモデルにも、同じ操作をします。

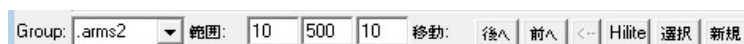
収納枠にも、背景色を付けることができます（明るい色が適しています）。因果関係には何の影響も与えませんが、視覚的にモデル構造を区別することができます。

#### 13.1.4 最小値と最大値と増分

方程式編集ダイアログでは、各方程式に、最小値、最大値、増分を入力できます。Vensim PLE 及び PLE Plus では、これらはわかりやすく表示されています。



他のラインナップでは、これらは [ 範囲 ] として表示されています。



ダイナミックなモデルでは、変数が大きすぎたり小さすぎたりした時、注意を告げるメッセージを表示するために、範囲は使用されます。例えば、範囲が [ 10,500 ] の時に、変数が 10 以下か 500 以上になると、警告が通知されます。

定数が変化できる幅も範囲で設定します。この場合、増分は、スライダの動く幅になります。例えば、範囲が [ 10,500,5 ] の場合、定数が 10 から 500 の範囲で、5 ずつ変化することを示しています。[ 0,1,1 ] の範囲は、スイッチのオン・オフに使用できます。

もし、定数にどんな範囲も指定しない場合、Vensim は、モデルの値を参考に自動作成します。また、増分を指定しない場合、範囲の幅を 80 で割った値が設定されます。

#### 13.1.5 モデル方程式

モデル方程式は、分かりやすいものです。範囲は、単位のあとに角括弧 [ ] で表示されます。「望ましい軍事力」を例外として、「~2」の方程式は、「~1」とまったく同じものであり、よってこれらの方程式は、ここでは表記しません。

初期経済力 1 = 100  
Units: 円/年 [10,500,5]

初期軍事力 1 = 50  
 Units: 円 [0,200,5]

必要となる軍事力 1 = MAX(0, 軍事装備の老朽化 1 + (望ましい軍事力 1 - 総軍事力 1)/軍事力の調整時間 1)  
 Units: 円/年

投資割当 1 = 0.3  
 Units: Dmnl [0,0.6,0.05]

投資効率 1 = 0.15  
 Units: 1/年 [0.01,0.4,0.01]

最大軍事割当量 1 = 経済力 1 \* 最大軍事割当 1  
 Units: 円/年

軍事力の耐用年数 1 = 20  
 Units: 年 [2,60,1]

軍事装備の老朽化 1 = 総軍事力 1 / 軍事力の耐用年数 1  
 Units: 円/年

軍事投資 1 = 最大軍事割当量 1 \* 軍事利用割当 1  
 Units: 円/年

経済力の衰退 1 = 経済力 1 / 経済力の耐用年数 1  
 Units: 円/年/年

経済力の耐用年数 1 = 30  
 Units: 年 [3,80,1]

望ましい軍事力比 1 = 1  
 Units: Dmnl [0,5,0.1]

経済力 1 = INTEG (経済力の成長 1 - 経済力の衰退 1, 初期経済力 1)  
 Units: 円/年

FINAL TIME = 100  
 Units: 年

軍事利用割当 1 = WITH LOOKUP (ZIDZ(必要となる軍事力 1, 最大軍事割当量 1),  
 [(0,0)-(10,1),(0,0),(1,1),(10,1)],  
 (0,0),(0.4,0.4),(2,0.8),(3,0.9),(5,1),(10,1)))  
 Units: Dmnl

経済力の成長 1 = 投資額 1 \* 投資効率 1  
 Units: 円/年/年

INITIAL TIME = 0  
 Units: 年

投資額 1 = 非軍事投資 1 \* 投資割当 1  
 Units: 円/年

最大軍事割当 1 = 0.4  
 Units: Fraction [0,1,0.05]

非軍事投資 1 = 経済力 1 - 軍事投資 1  
 Units: 円/年

SAVEPER = TIME STEP  
 Units: 年 [0,?]

望ましい軍事力 1 = 総軍事力 2 \* 望ましい軍事力比 1

Units: 円

望ましい軍事力 2 = 総軍事力 1 \* 望ましい軍事力比 2

Units: 円

TIME STEP = 0.125

Units: 年 [0,?]

軍事力の調整時間 1 = 5

Units: 年 [1,20,1]

総軍事力 1 = INTEG (軍事投資 1 - 軍事装備の老朽化 1, 初期軍事力 1)


Units: 円

## 13.2 構造と振る舞いとスピード

統合シミュレーションは、モデルの構造と振る舞いを、統合してシミュレーションできます。出来事と振る舞いと構造は階層として考えられます。つまり構造が振る舞いを生み、振る舞いのパターンによって出来事が生じるという見方です。

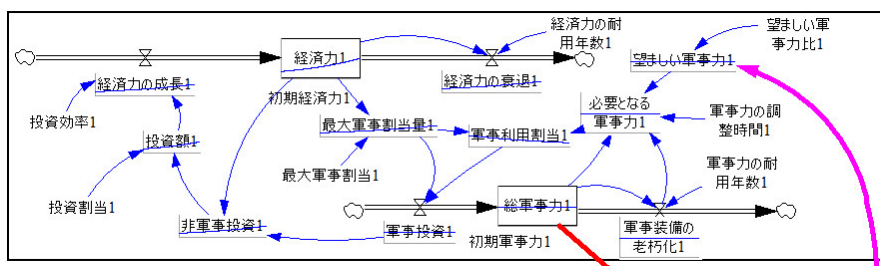
シミュレーションモデルは構造を表し、シミュレーションを通じた振る舞いを生みます。モデルの構造と振る舞いは、強い関連性を持ちます。しかし、モデルの構造が、どのようにして振る舞いを引き起こすか、理解するのは難しいかもしれません。構造に、振る舞いを重ね合わせて、直ちにその振る舞いを更新するものが、統合シミュレーションです。

### 13.2.1 振る舞いの表示

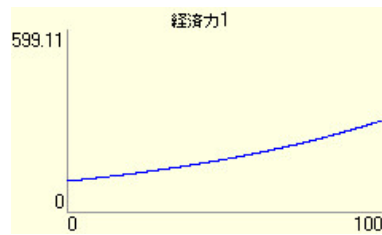
➤ シミュレーションの実行  をクリックします。

➤ [表示] メニュー [グラフ簡素表示] を選びます。または、**[B]**を押します。

以下ようになります。




ダイナミックな変数の時間による振る舞いが、スケッチに直接表示されています。スケッチ上で、振る舞いを表示しながら、変数や矢印を移動したり、方程式を変更することができます。カーソルを変数の上に移動し、しばらくすると、グラフが拡大して表示されます。



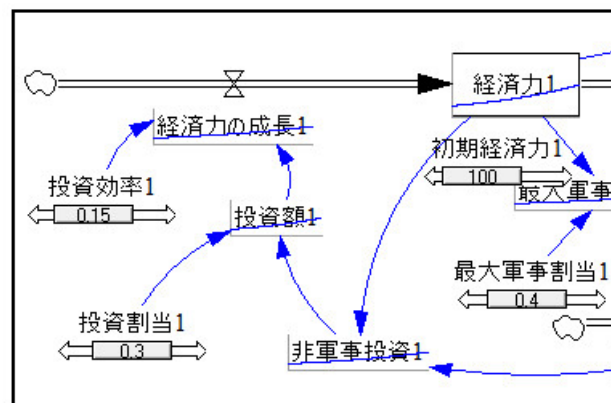
グラフにはラベルが追加されています。内容は、スケッチに表示されたサムネイルグラフと同じです。変数をクリックし、ワークベンチ変数とすることもできます。

多くのシミュレーションを作成した場合、それらはサムネイルグラフ上で、それぞれ異なる色で表示されます。制御パネルからデータセットを管理することによって、グラフに表示されるシミュレーション結果を操作できます。この例は、後で紹介します。

### 13.2.2 シミュレーションの作成

- 統合シミュレーションの実行  をクリックします。
- [データセット Current はすでに存在します。上書きしますか?] と表示されるので、[はい] をクリックします。

変数上のグラフに加えて、定数にスライダが表示されます。



すべての定数の名前のすぐ下に、自動的にスライダが作成されます。形状を持つ定数で、その名前が下に配置されている場合は、名前の上にスライダが作成されます。統合シミュレーションの実行中は、定数や表関数に変更を加えるたびに、シミュレーションされ、結果が反映されます。こうした変更には、いくつかの方法があります。

#### スライダの移動

スライダのボタンを動かすことで、定数を変更することができます。

- 「望ましい軍事力比 1」の下にあるスライダへカーソルを移動します。
- スライダを、マウスで押します。

ボタンが沈みます。スライダの今の位置を分かり易くするために、ポインタが少し動きます。


- スライダの値が「2」になるまで、ゆっくり右にドラッグします。

スライダを移動しながら、モデルのストックがどう変化しているか確認します。水平を示している総軍事力（総軍事力1と2）のグラフが、だんだんS字状に振る舞います。グラフは、増加を示す場合は赤で、減少を示す場合は青で点滅します。

もし何か手順が抜けていたら、バックアップして、再度試してください。


- マウスを放します。

移動させていたスライダは、グレーの元のスライダに戻ります。

- **Home** を押すか、最新の変更をリセット  をクリックします。

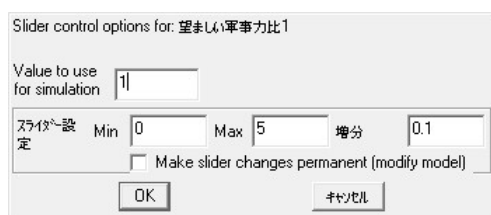
このスライダの値は、モデル値に戻り、元の初期条件でもう一度シミュレーションされます。軍事力のグラフが水平になるのを確認してください。

### 13.2.3 スライダ値の設定

スライダは、定数の方程式で設定された範囲の中で動きます。もし、範囲が設定されていなければ、定数の値に応じて自動計算された範囲で動きます。また、スライダのルール  をクリックすることで、定数の値とその範囲の両方を設定することもできます。

- 「望ましい軍事力比1」の下にある、スライダのルール（スライダそのものではありません）をクリックします。

ダイアログが開きます。間違えてスライダのボタンをクリックした場合は、マウスを放して、もう一度やり直してください。




- [ Value to use for simulation ]【シミュレーションに使われる値】を「2」とし、[ Min ]【最小値】を「1」、[ Max ]【最大値】を「2」、[ 増分 ]を「0.01」とし、[ OK ]をクリックします。

「望ましい軍事力比1」を「2」として、モデルがシミュレーションされ、結果が表示されます。スライダは、右端にあります。


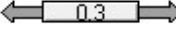
- スライダを動かして、結果を検討してみます。




スライダの値は、より細かく操作できるようになっています。

- **Home** を押すか、最新の変更をリセット  をクリックします。

「望ましい軍事力比 1」の値が「1」にリセットされます。しかし、スライダの幅は、変更したときのままです。スライダの幅は、表示画面を切り替えるか、統合シミュレーションを停止するまで変更されません。範囲を設定する時に、ダイアログにある [ Make Slider Changes Permanent (modify model) ]【スライダの変更を固定する(モデルを修正する)】にチェックを入れた場合、モデルそのものが変更されます。

### 13.2.4 矢印キーの利用

マウスを利用してスライダの値を変更するだけでなく、矢印キーも使用できます。統合シミュレーションの実行中に、他とは異なって見えるスライダが 1 つあります。ほとんどのスライダはこのような  見えますが、その中の 1 つは、スライダのレールが  濃く表示されています。このように、レールが濃く表示されているものを、アクティブスライダといいます。スライダをクリックすることで、アクティブスライダになります。**Tab** (もしくは **Shift** + **Tab**) を押して、スライダ間を移動することもできます。しかし、定数を作成した順に移動するため、分かりにくいこともあります。

- 「望ましい軍事力比 1」の下のスライダをアクティブにします。もし選択されていなければ **Tab** でアクティブにします。
- 右矢印  を押します。  
スライダの値が「1」から「1.01」になり、グラフに少し変化があります。もう一度  を押すと、値は「1.02」になります。
- 左矢印  を押します。

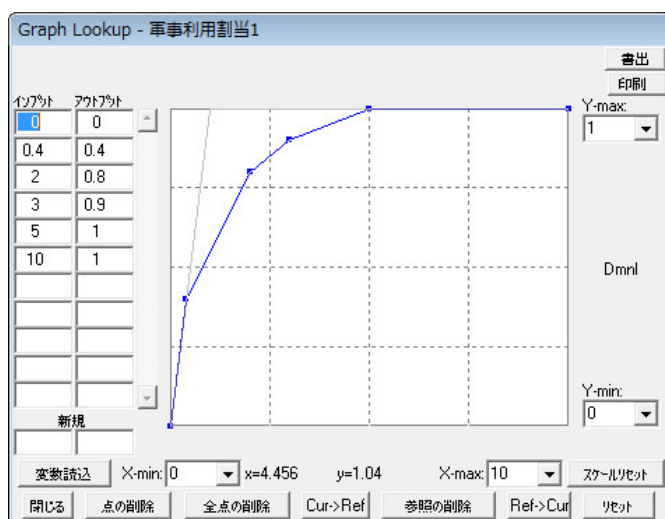
スライダの値が「.01」ずつ減少します。矢印キーは、定数の値を少しずつ動かしたい場合に、とても便利です。また、規模の大きいモデルにおいて、定数の値を変更するごとに、1 つだけシミュレーションを実行したい場合にも有効です。

### 13.2.5 表関数の変更

表関数の変更は、表関数編集ダイアログを開いて、修正します。モデルに修正を加えると、シミュレーションが実行され、その結果が表示されます。

- 「望ましい軍事力比 1」の下のスライダをドラッグし、値を「2」にします。
- 「軍事利用割当 1」をクリックします。

表関数編集ダイアログが開きます。



統合シミュレーションを実行中の表関数編集ダイアログでは、[ 閉じる ] ボタンだけではなく、[ リセット ] ボタンがあります。表関数編集ダイアログで値を変更することにより、モデルはシミュレーションされるため、変更を取り消すことはできません。[ リセット ] は、モデルの値を元に戻します。



この表関数は、最小値と最大値から得られた参照線（グレーで表示されています）を持ち、グラフはこの参照線上か、もしくはその下側に位置されます。





- マウスを、グラフの 3 つめの点の上に移動させ、この点を参照線までドラッグします。

点を動かすことによる変数の振る舞いを見ます。他の変数が隠れてしまう場合は、表関数編集ダイアログのタイトルバーをドラッグすることもできます。

- [ 閉じる ] をクリックし、ダイアログを閉じます。


### 13.2.6 ツールバーでの変更（PLE と PLE Plus は対応しません）

定数と表関数の変更は、モデル定数の変更  や、表関数の変更  でも可能です。これらは、定数や表関数のリストを持つダイアログを表示します。


- モデル定数の変更  をクリックします。
- リストから、「望ましい軍事力比 1」をクリックし、[ 変更 ] をクリックするか、 を押します。編集ボックスに、現在の値が強調表示で示されます。
- 「1.8」と入力し、 を押します。モデルがシミュレーションされます。「望ましい軍事力比 1」の下のスライダの値が変更されています。
- [ 閉じる ] をクリックします。
- 表関数の変更  をクリックします。
- 「# 軍事利用割当 1 #」（# は、Vensim によって追加されています）をクリックし、[ 変更 ] をクリックします。
- 先程のように表関数を変更し、[ 閉じる ] をクリックします。

### 13.3 シミュレーション実験

ここまで、変数が 1 つのグラフだけを見てきました。特に重要なシミュレーションは保存することができます。また、サムネイルグラフにいくつかのシミュレーション結果を表示することもできます。

- 定数/表関数の変更を全てリセット  をクリックするか、**[Ctrl]+[Home]**を押します。

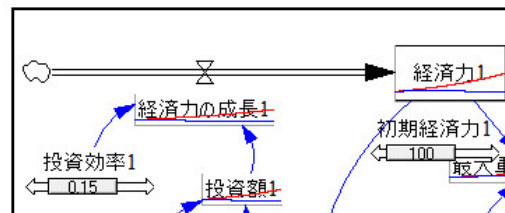
このリセットで、定数だけでなく、表関数の変更もリセットされます。




- 実行結果の保存  をクリックします。
- 「baserun」と入力します。
- [保存] をクリックします。

画面が点滅しますが、変化がないように見えます。これは、「baserun」がこれまでの実行結果と同じだからです。

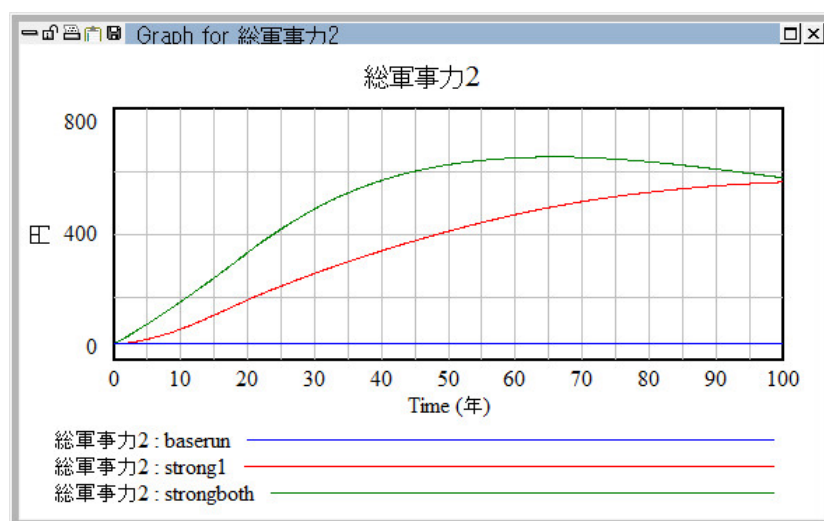
- 「望ましい軍事力比 1」のスライダを移動させ、「2」とします。

各変数に、2 つのグラフが表示されます。




- 実行結果の保存  をクリックします。「strong1」と入力し、[保存] をクリックします。
- 「望ましい軍事力比 2」のスライダを移動させ、「2」とします（この変数は、モデルの下段にあります）。
- 実行結果の保存  をクリックします。「strongboth」と入力し、[保存] をクリックします。
- 「総軍事力 2」をクリックし、グラフ  をクリックします。以下のようになります。






*strong1* と *strongboth* が、どのようにほぼ同じ値に達するのかがわかります。

➤ 「経済力 2」をクリックし、グラフ  をクリックします。

両国が軍事力強化に積極的になるほど、経済活動と同様に、軍事力も停滞します。望ましい軍事力の値を操作してシミュレーションを行うと、100 年後の軍事力が最も高いのは、その値が「2」よりも少し低い値だと分かります。

### 13.3.1 実行結果の読み込みと取り除き

制御パネルから、データセットの読み込みと取り除きや他の設定の変更ができます。時間軸の設定を変更した場合、サムネイルグラフには反映されませんが、グラフの表示が変わります。

➤ 制御パネル  をクリックし、[データセット] タブをクリックします。

➤ 「*strong1*」をクリックし、移動 << ボタンをクリックします。


➤ 「*baserun*」をクリックし、移動 << ボタンをクリックします。

これらの変更によって、サムネイルグラフは読み込まれたデータセットを反映して再表示されます。16 個までのデータセットを読み込むことができますが、多くのデータセットを読み込んで役に立つことはめったにありません。

## 13.4 フィードバックループの遮断

モデルのシミュレーションによる振る舞いを理解する効果的な方法は、フィードバックループを遮断し、その変化を見ることです。統合シミュレーションでは、こうしたことが簡単にできます。

### 13.4.1 競争の中止

- 定数/表関数の変更を全てリセット  をクリックするか、**[Ctrl]+[Home]**を押します。

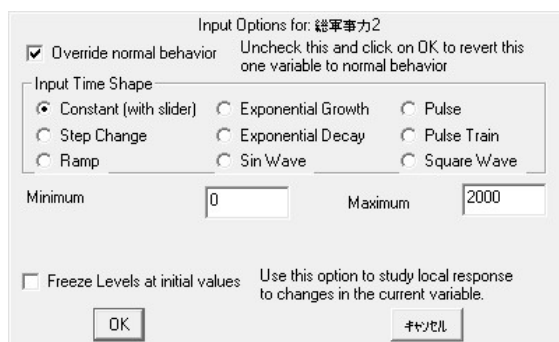
これで、すべてのスライダを元の位置にリセットします（表関数もリセットされます）。もしスライダが 1 つしかなく、そのスライダがアクティブである場合、現在のスライダをリセットするのと同じです。

- 「望ましい軍事力比 1」を「2」とします。

2 本のグラフが表示されます。表示されない場合は、「*strong1*」と「*current*」が読み込まれているか確認してください。

- 「総軍事力 2」を右クリックします（または、**[Ctrl]**を押しながらクリックします）。


変数の優先ダイアログが表示されます。



変数に優先が設定されると、シミュレーションで計算された値ではなく、外部から入力された値が利用されます。この値は、定数またはパターンを持つ値です。

- [Maximum]【最大値】に「1000」と入力します。
- [OK] をクリックします。

モデルが「総軍事力 2」の定数 500 でシミュレーションされます。通常の方法で計算されていないことを示すために、「総軍事力 2」は異なる色で表示されます。また、その下にスライダが表示されます。

- 「総軍事力 2」の下のスライダをドラッグし、値を 50 に下げます。これは、この変数の初期値です。
- 「総軍事力 1」をクリックし、グラフ  をクリックします。

「総軍事力 1」が 50 で開始され、すぐに 100 に達し、そのままの値を保ちます。進行中の増加は見られません。

- ツールバーにある、優先の中止  をクリックします。

1 本の線のグラフが表示されます。

### 13.4.2 振る舞いの優先

このモデルでは経済力は、軍事力を生み出す能力に大きな影響を及ぼします。軍事投資がその能力に応じて決まるのであれば、その値を用いることができます。しかしながら、実際の軍事投資が、経済力を超えることはないので、このような定式化は困難です。このモデルにおいて、経済活動による制限をなくす適切な方法は、定数を作成するのではなく、軍事投資に影響を受けない経済活動をモデルに組み込むことです。

- 「経済力 1」を右クリックします（もしくは $\text{Ctrl}$ を押しながらクリックします）。
- [ Exponential Growth ] のラジオボタンをクリックします。
- [ Starting From ] 【初期値】を「100」とします。
- [ Growth Rate ( %/unit time ) ] 【増加率】を「10」とします。

他の変数の振る舞いを見えます。「総軍事力 1」がさらに増加しますが、まだ S 字を描いています。実際、「経済力 2」は *strong1* の実行結果よりも大きく低下し、その軍事力を制約しています。同じように、「経済力 2」も変更し、「経済力 1」への影響を見ます。

- 「経済力 2」を右クリックします（もしくは $\text{Ctrl}$ を押しながらクリックします）。
- [ Exponential Growth ] のラジオボタンをクリックします。
- [ Starting From ] を「100」とします。
- [ Growth Rate ( %/unit time ) ] を「10」とします。



シミュレーションによる振る舞いを見ます。両国共に、軍事力が急速に増加しています。しかし経済力が抑制する要因ではないことが分かります。なぜなら、どちらの国でも、軍事力に対する経済力の比が、減少しているからです。

- 「望ましい軍事力比 2」の値を「2」にします。

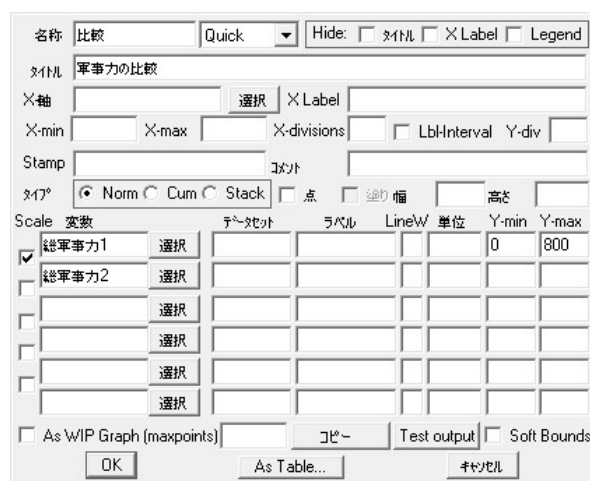
経済成長は、年 10 % の成長率であっても再び抑制されます。


## 13.5 制御パネルの作成



総合シミュレーションでは、サムネイルスケッチによるグラフに加えて、グラフと表を入出力オブジェクトとして操作することができます。

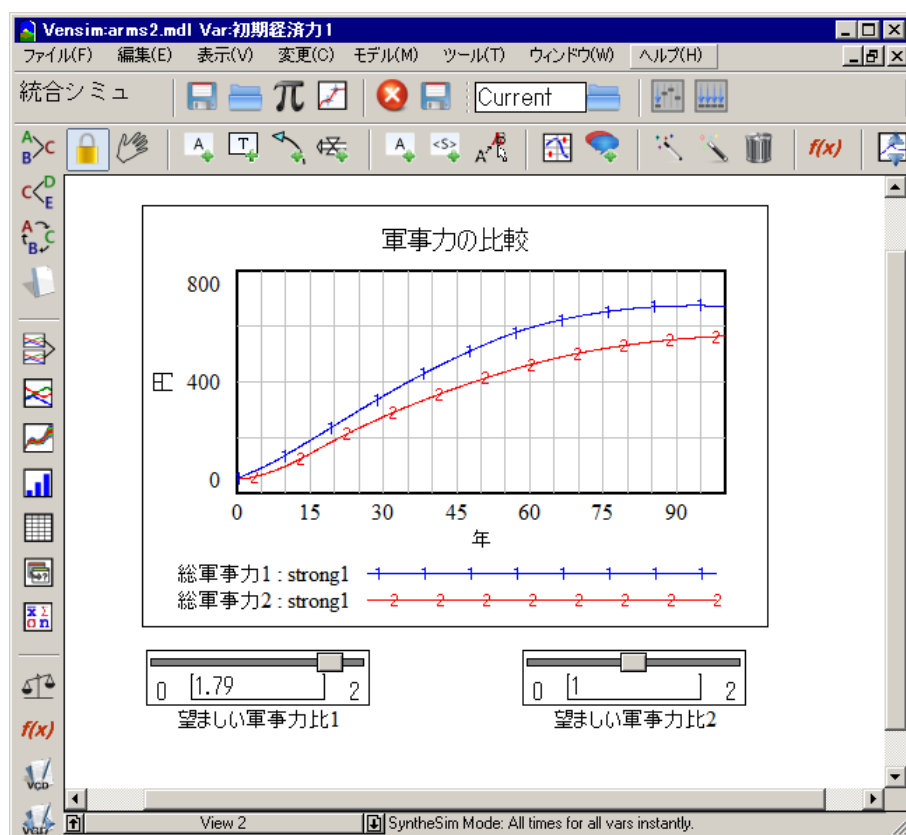
- シミュレーションの停止  をクリックします。
- 制御パネル  をクリックし、[ グラフ ] タブをクリックします。
- [ 新規 ] をクリックします。
- カスタムグラフダイアログの [ タイトル ] を「軍事力の比較」とし、変数に、「総軍事力 1」と「総軍事力 2」を追加します。[ 名称 ] を「比較」と入力します。
- 変数名の左側の [ Scale ] 【目盛り】のチェックボックスにチェックを入れます。一番上の行の [ Y-min ] に「0」、[ Y-max ] に「800」と入力します。
- [ Soft Bounds ] にチェックを入れます。ダイアログは以下のようになります。

[ Soft Bounds ] は、グラフの Y 軸を自動調整する機能です。



- [ OK ] をクリックします。
- [ 表示 ] メニュー [ 新規ファイル ] を選択します。
- 入出力オブジェクト  をクリックし、スケッチ上でクリックします。
- ダイアログの中の、[ 自作グラフ出力 ] のラジオボタンをクリックします。[ 自作グラフ/分析ツール出力選択 ] のドロップダウンボックス [ ] から、「比較」を選択し、[ OK ] をクリックします。
- グラフを大きくします。
- グラフの左下をクリックします。
- 入出力オブジェクトのダイアログで、[ 定数... ] をクリックし、[ 望ましい軍事力比 1 ] を選択します。
- [ スライド設定 ] の [ 範囲 ] で、[ From ] を「0」[ to ] を「2」[ 増分 ] を「.01」とし、[ OK ] をクリックします。
- グラフの右下をクリックします。
- 入出力オブジェクトのダイアログで、[ 定数... ] をクリックし、[ 望ましい軍事力比 2 ] を選択します。
- [ スライド設定 ] の [ 範囲 ] で、[ From ] を「0」[ to ] を「2」[ 増分 ] を「.01」とし、[ OK ] をクリックします。

スライダを少し大きくし、位置を調節します。制御パネル  ですべてのデータセットを削除します。次に、統合シミュレーション  を実行し、スライダを移動させます。以下ようになります。



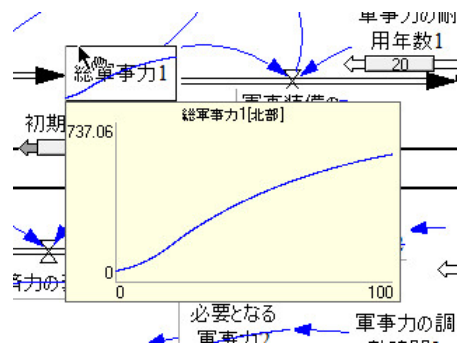
先程の表示画面に戻って、モデルに変更を加えることもできます。再度この表示画面に戻った時に、その結果をグラフに見ることができます。

### 13.6 下添え字と統合シミュレーション (DSS と Professional 版のみ対応)

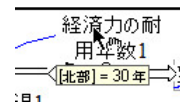
下添え字と統合シミュレーションについて簡単に紹介します。統合シミュレーションでは、変数に対して1つのグラフが表示され、定数にはスライダが表示されます。Vensimは下添え字制御ダイアログで、表示する下添え字が管理されています。選択されている下添え字の最初のものが表示されます。

グラフにおいて、どの値(下添え字)が表示されているのかを確認するには、グラフの上でマウスを動かして表示されるポップアップグラフで確認できます。例えば以下のグラフは、「総軍事力1」に下添え字が設定され、その「北部」の値が表示されています。

下添え字は17章で説明します。これは配列を意味し、ある国が北部と南部から成るといった場合などに使います。



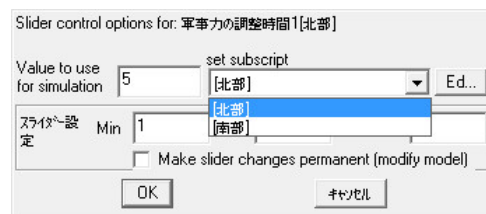
また定数についても、その変数名上でマウスを動かして表示されるポップアップがあります。



#### 注意

この場合スライダではなく、変数名上でマウスを動かします。

スライダのハンドルをクリックすることで、別の下添え字を参照させることもできます。以下のように表示されるダイアログで、下添え字を変更します。



別の下添え字を設定します。新しい下添え字がスライダに反映されます。また [Ed..] ボタンは、下添え字の全ての定数を変更することができます。

## 第 14 章

# リアリティチェック

問題を検討するために、モデルは作成され、利用されます。モデルを作成したのち、その現実性を確認するために必要なチェックがあります。これらのチェックは、明示的で、その振る舞いなどを見るものがある一方で、メンタルモデルとの比較やモデル作成過程における検討があります。いずれの場合であっても、これらのチェックは大変重要で、作成したモデルが実際に応用される課題に適用可能であることを保証するうえで重要です。

リアリティチェックは、モデルにおいて成立するはずの条件文を作成し、その条件文を自動的に検証します。リアリティチェックは、作成したモデルを検証するための新しい技術です。

本章の内容は以下の通りです。

- リアリティチェックの概念を紹介します。
- 制約とテスト入力を定義する方法を示します。
- リアリティチェック方程式への適合性をテストする方法を示します。
- リアリティチェックを使用した例を示します。

### 14.1 モデルと現実性

モデルは現実を表現するものであり、現実の認識でもあります。モデルの有効性を検証するには、現実が、モデルの中でも再現されているかどうか重要です。この検証は、測定値とモデルの振る舞いを比較することで行われます。この比較は時系列データの振る舞いを対象に、その変化だけでなく、モデル条件を変化させた感度テストなどで検討します。

モデルの妥当性を検証する方法には、そのモデル構造についての検討もあります。モデルのユーザーは、意思決定を行う上で、わからない情報は必要としません。因果関係をはっきりさせておく必要があります。

モデルの構造とその振る舞いには、滅多に生じないケースを考えることができます。つまり“もし\_\_\_\_となれば\_\_\_\_となる”という文ができれば、モデルの問題を見つけて、モデルの信頼性を高めることができます。

ほとんどの場合、モデルが妥当であることを検証する条件文は非常に重要です。その検証の多くは、完全なモデルを使うのではなく、モデルの一部分や場合によっては、関係式にもとづき行われます。このとき、モデル作成者の経験が必要になります。

しかしながら、これまで行われてきたモデルの検証は、十分であったとはいえません。

なぜなら、従来のモデル検証は、モデルの一部を切り取ったテストであったり、異なる入力値による振る舞いを見たり、基本のモデル構造を変更したり、多くのシミュレーションを行っているに過ぎません。こうした検証を行ったとしても、モデルの修正が必要となったり、その修正効果はテストされないためです。

リアリティチェック方程式はモデルが妥当であるために必要な条件文を作成します。また自動的に、こうした条件文の適合性を検証します。作成した条件文は、モデルのバージョンには縛られません。なぜならこうした条件文は、通常のモデル方程式とは区分されていて、モデルの通常の関数には影響しないためです。またモデルに与えた条件式に反しているかどうかを簡単に確認することができます。

#### 14.1.1 専門家の役割

リアリティチェック方程式は Vensim モデリング言語を拡張したものです。リアリティチェック方程式を書くために要求されるスキルと経験は、モデルを描くために要求されるものとは異なります。リアリティチェック方程式は、現実の振る舞いを対象にします。それらは、振る舞いを発生させる構造を必要とはしません。リアリティチェック方程式は、振る舞いの条件を作成し、モデルが条件に従った反応を行っているかどうかをチェックします。

リアリティチェックは、振る舞いだけを対象にしています。したがって条件式の作成は、こうした振る舞いに対して専門的な知識を持つ人が適しています。専門家の知識を入れることで、モデルのバグを直すよりも、有効な検証ができます。そのことでモデルの信頼性が高まります。

### 14.2 リアリティチェック方程式の定義

Vensim に方程式を入力するのと同じ方法で、リアリティチェック方程式は入力します。リアリティチェック方程式の入力には、スケッチツールを使用します。また方程式エディタやテキストエディタを使うこともできます。リアリティチェック方程式はモデル方程式ではありません。しかし、その定義には、モデル変数を使用します。

繰り返しますが、リアリティチェック方程式は因果関係を示す条件式ではなく、振る舞いを示すものであることに注意してください。つまり“このことが生じれば、こうなるであろう”という振る舞いの関係です。

リアリティチェック方程式の命名規則は、モデル変数と異なります。モデル変数は、わかりやすい名詞が適しています。例えば、「労働力」、「生産性」、「能力」、「決定」などです。これに対して、リアリティチェック方程式は、チェックの本質を示す簡潔な文章が適しています。例えば「労働者なければ生産なし」、「雨は洪水をもたらす」のようにします。リアリティチェック方程式に名前を付けるための良い方法は、それらが正しいか誤りかを考え、その条件文が成立する場合の文にします。

リアリティチェックを行ううえで、2 種類の方程式があります。それは制約とテスト入力です。制約は、与えられた条件から得られる結果についての条件文を作ります。制約と言われるのは、テスト入力がある振る舞いを制約する条件を特定するためです。制約の違反は、モデルに問題があることを意味します。テスト入力は、制約をもたらす条件や状況を



特定する方法です。テスト入力に制約で利用されるので、最初に説明します。

### 14.2.1 テスト入力

テスト入力はモデル変数の方程式を変更することによって、代わりの条件文を定義することができます。テスト入力の基本的な形式は以下のとおりです。

```
name :TEST INPUT: variable = expr
```

[ name ] はテスト入力の名前です。「TEST INPUT」の右側には、変数の条件式を入力します。この変数はモデル変数です。

条件式では、動的な関数 (SMOOTH など) や定義された関数やマクロなどは利用できません。こうした機能が必要な場合は、テスト入力に用いることができるモデル変数を作成します。

テスト入力を定義する理由は、検討する条件に名前をつけることです。これによって制約がよりわかりやすくなります。

#### 動的なテスト入力

テスト入力には、一定時間後に値を変化させることが有効な場合もあります。例えば、時間が 10 から 12 の間に、生産を 0 に減少させることを考えます。この種の変化は、リアリティチェックを完成したり、テスト入力に対するモデルの振る舞いを検討したりする上で役立ちます。

時間的な変化を持つテスト入力には、RC で始まる関数を使います。例えば、RC COMPARE、RC DECAY、RC GROW、RC RAMP、RC STEP があり、その詳細はリファレンスマニュアルの 4 章にあります。これらはすべて以下のように定義します。

```
生産をゼロに:TEST INPUT:
生産 = RC RAMP(生産,0,2,10)
```

このテスト入力は、時間が 10 から 12 までの間、生産を 0 に減らします。

RC で始まる関数は全て、開始時間と継続時間という 2 つのオプションを持ちます。もし継続時間が省略された場合、テスト入力の変更された値はシミュレーションの最後まで利用されます。また継続時間が入力されていれば、その後は元の値に戻ります。

もし開始時間 (上記の例では 10) が省略され、モデルが定数 RC START TIME を含む場合、指定された変更はこの時間に開始します。もし、RC START TIME がモデルにない場合、変化は INITIAL TIME + TIME STEP に開始します。

RC START TIME は便利に利用できます。なぜなら、変化をもたらす時間を設定する一方で、RC 関数での細かい設定を省略できるためです。シミュレーションの途中でテスト入力を変化させることは、モデルのテストにおいて有効です。

### 14.2.2 制約

制約は以下のようになります。

```
name : THE CONDITION : condition : IMPLIES : consequence
```

:*THE CONDITION* :と: *IMPLIES* :は、Vensim の特別なキーワードです。condition と consequence は、以下に示すような論理式です。制約の名前は、Vensim のその他の変数と同じように、文字と数字を使用します。制約は、計測の単位を必要としません。しかしテキストエディタを使用する場合、その式ではチルダ ( ~ ) を使用します。また制約にコメントを付けることもできます。

#### 論理式

制約は、条件と結果を使用します。それらは論理表現であり、以下のようなものです。

資本がなければ生産なし : *THE CONDITION* : 資本 = 0 ;  
: *IMPLIES* : 生産 = 0

リアリティチェック方程式をテストした場合、条件が成立する場合に、結果も成立するかどうかをチェックします。条件が成立していても、結果が成立しない場合は、リアリティチェックが失敗したことになります。

論理式では、=、>、<、<>、: *OR* :、: *AND* :、: *NOT* : などを用いてより複雑なものも作成できます。

人口 > 8E9 : *AND* : (食料割合 < .75 : *OR* :  
人口 > 限界人口)

この式では多くの事が比較されています。つまり、「人口 > 8E9」であり、なおかつ、「食料割合 < .75」であるか、または「人口 > 限界人口」が成り立つ場合、この条件が成立することになります。論理式は、一見して理解することは難しいので、あまり多くの条件を与えないようにします。結果についても、多くの項目を: *AND* : で結び付けることができますが、あまり複雑にすると役に立ちません。

制約における条件式は、変数間の比較もしくは変数と数値の比較に限定されています。唯一の例外は、var=expr もしくは、テスト入力の利用です。

したがって、以下は、var < expr となっているために間違いです。

扶養量よりも多い人口 : *THE CONDITION* : 人口 < 扶養量 \* 1.1  
: *IMPLIES* : 過密による死亡 < 1000

var=expr を用いた正しい条件式は以下の通りです。

扶養量よりも多い人口 : *THE CONDITION* : 人口 = 扶養量 \* 1.1  
: *IMPLIES* : 過密による死亡 < 1000

この方法で、制約の条件に条件を含むには、*TIME TRANSITION* を使用し、テスト入力の規則に従わねばなりません。

テスト入力は、以下のように、制約の条件に使用されます。

扶養量よりも多い人口 : *THE CONDITION* : 扶養量よりも多い人口の加算  
: *IMPLIES* : 過密による死亡 < 1000

「扶養量よりも多い人口の加算」はテスト入力です。テスト入力は、それが成立するならば有効ならば真を、そうでないならば偽の値をとる論理変数とみなされることに注目してください。

制約の条件文では、関数 ( MIN、MAX、SUM など ) の使用が制限されています。しかし結果の条件文では制限がありません。

結果の同一性をテストすることはあまり有効ではありません。なぜなら計算方法によって、値の微妙な違いが生じることがあり、そのために同一と見なされないことがあるためです。

### 14.2.3 結果における動的なテスト

RC 関数をテスト入力に持つリアリティチェック方程式は、通常、結果で RC...CHECK 関数を用います。RC...CHECK 関数はテスト入力における RC... 関数とよく似ています。テスト入力を変数の値を変更させた際に、それがもたらす結果を比較します。RC...CHECK 関数は、RC... 関数よりも引数が増えます。それは猶予期間であり、結果が反映されるまでの遅れを考慮するものです。例えば以下のようになります。

```
生産をゼロ:TEST INPUT:
    生産 = RC STEP(生産,0)
生産なければ出荷なし:THE CONDITION:生産をゼロ:IMPLIES:
    販売 <= RC RAMP CHECK(.5, 販売,.0001)
```

テスト入力は生産をゼロにします。その 0.5 時間後（猶予時間）に「販売」がそれまでの販売を 0.0001 倍したものよりも少ないかどうかをチェックします。ゼロではなく、0.0001 を用いるのは、連続値を使った方が、エラーが生じにくいからです。

猶予期間は RC...CHECK 関数の第 1 引数になります。

### 14.2.4 :CROSSING:

リアリティチェックにおける関係性において、:CROSSING: や:AT LEAST ONCE: が以下のように用いられます。

```
... :IMPLIES:
    在庫 > :CROSSING: RC STEP CHECK(0, 在庫,1)
```

もし在庫が基準値より急増し、その後少なくなり、そのまま推移するのであれば、:CROSSING:が 2 つ必要になります。

```
在庫 > :CROSSING: :CROSSING: RC STEP CHECK(0, 在庫,1)
```

:CROSSING:を用いる場合、最後に:IGNORE: を使うことによって、その後の交差を無視することになります。

したがって以下の場合、在庫は当初増加し、その後減少しますが、その後のことは考慮しません。

```
在庫 > :CROSSING: :IGNORE: RC STEP CHECK(0, 在庫,1)
```

### 14.2.5 :AT LEAST ONCE:

:AT LEAST ONCE:は、:CROSSING:と似ています。これは、条件が 1 度だけ真になることを意味します。

つまり以下の条件文は、「在庫」がそれまでの値よりも少なくとも 1 度は増加することを示しています。

```
在庫 > :AT LEAST ONCE: RC STEP CHECK(0, 在庫,1)
```

### 14.2.6 空の条件

条件において、制約を空にすることもできます。

負債限界:THE CONDITION: :IMPLIES: 負債 < 4E6

これはどんなことが起きようとも、負債が 4 百万を越えないことを意味しています。

この例では、負債の最大値を 4 百万に設定し、それを越えた場合はメッセージを表示します。

### 14.2.7 結果における未知のテスト

すべての変数で条件を満たすかどうかをテストするには、変数名ではなく、ワイルドカードを用います。

全世界の平和:THE CONDITION: FINAL TIME=101 :IMPLIES:  
\* < 1E9 :AND: \* >= -1E3

これはすべての変数が、1000 から 10 億の間にあるかどうかをチェックしています。条件に「Final TIME=101」としたのは、空の条件であれば時間を要するためです。

## 14.3 シミュレーションとリアリティチェック

リアリティチェック方程式を実際に操作する前に、モデルの中で何が起きているのかを簡潔に説明します。リアリティチェック方程式はモデルの基本構造に組み込まれます。それらは、影響の因果関係が定義されていないので、感度分析とは質的に異なります。テスト入力と制約は、どの時点であってもモデルに変化をもたらすことができます。

リアリティチェック方程式の実行を反映するため、モデルは再構築され、結果が成立するかを計算します。リアリティチェック方程式の計算後、元のモデル構造に戻ります。つまり、実行結果に対する因果関係の分析では、テスト入力による意外な結果をもたらすことがあります。

時に、モデルの再構築は、モデルを不適切にするかもしれません。最も多い問題は、モデルが同時方程式を持つため、値を計算できないことです。このような場合、Vensim は問題を報告し、シミュレーションは行われません。振る舞いの関係だけを考慮するリアリティチェック方程式では、シミュレーションのための方程式は必要ありませんが、シミュレーションには方程式が必要です。可能であるならば、シミュレーション方程式の問題を発生させないようにテスト入力を変更します。

注意 リアリティチェック方程式は、インタープリタ（1 行処理）として実行されます。そのため方程式は、連続的な再構築が必要です。

### 14.3.1 実行中の制約チェック

制約テストの実行中、制約の条件文が真であるとして、変数の値やモデル構造を構築します。

- 条件が同じであるか、またはテスト入力であれば、モデル方程式にリアリティチェック方程式を追加します。新しい変数値は、どこでも使えることになります。
- 条件が同じでなければ、まずその違いが真であるかどうかをテストします。
  - － もし真ならば、変数の値はそのままです。
  - － もし偽ならば、同じ条件の場合の値を割り当てます。

条件が真となれば、その結果も真になるかどうかテストします。

### 14.3.2 受動的な制約チェック

受動的な制約チェックでは、制約方程式の半分が論理式として評価されます。もし条件が真であるにもかかわらず、結果が偽となればエラーを報告します。

リアリティチェックでは、明示的に対象としなかった変数は受動的な適合性がテストされます。このチェックは通常のシミュレーションでは実行されません。

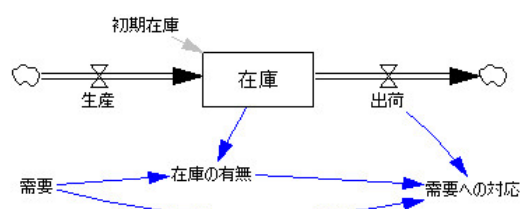
### 14.3.3 エラーのリポート

制約の違反は、まずエラー報告されます。そして適合性に問題があることを示します。

## 14.4 リアリティチェック方程式の入力

リアリティチェック方程式は通常の方程式と同じように入力します。テキストエディタでは、それを入力するだけでよいです。スケッチエディタでは、テスト入力や制約に利用する要素を描く必要があります。また、テスト入力における右項の式には、原因となる変数は含まれません。したがって、方程式は以下ようになります。

```
在庫 = INTEG (生産 - 出荷, 初期在庫) ~~|
在庫の有無: TEST INPUT: 在庫 = 3 * 需要 ~~|
需要への対応: THE CONDITION: 在庫の有無
: IMPLIES: 出荷 >= 需要 ~~|
```




リアリティチェックで用いられる因果関係や方程式は、通常のモデルとはわけた方がわかりやすいです。そのためには、グループを用いたり、別の表示画面を作成します。

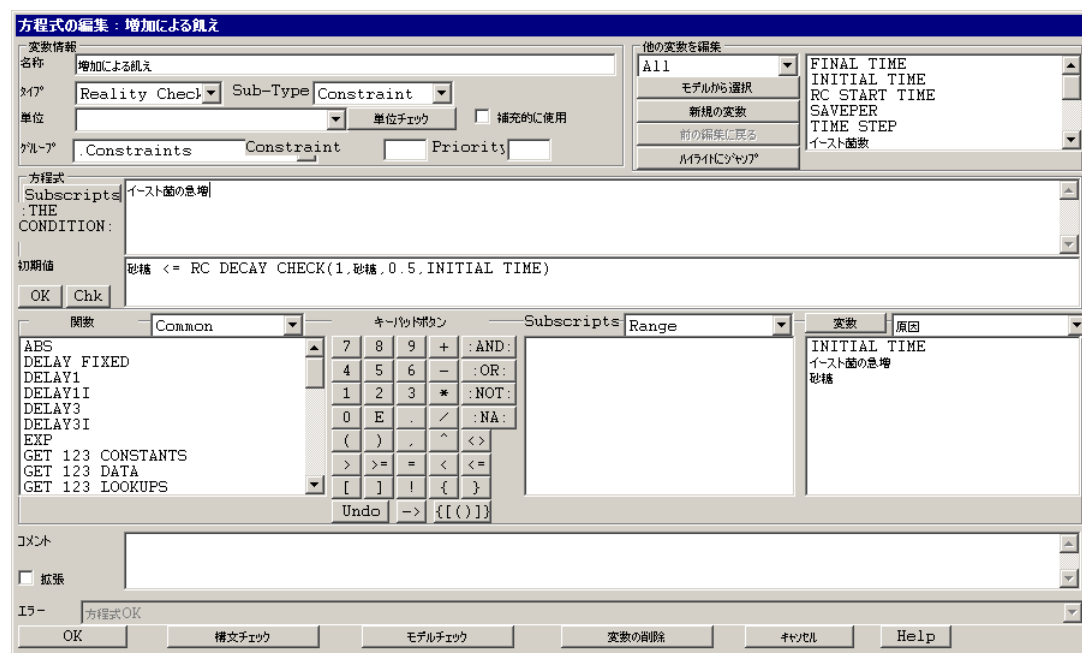
この図では、こうした変数が混在しています。「需要への対応」への矢印は、因果を示すものではなく、その真偽を知るためのものです。したがって因果関係の図と混在するため理解が難しくなります。したがってリアリティチェックでは、モデル構造との混乱を避けるために専用の表示画面を作成します。

Test Input と Reality Check  
の頭文字

またリアリティチェックの名前を工夫することで、よりわかりやすいものになります。  
例えばテスト入力には、変数名に “TF” を制約には “RC” を付けることがあります。

#### 14.4.1 方程式エディタ


リアリティチェックは、通常の方程式と同じように入力します。まず変数を作成し、つぎに方程式  で方程式編集ダイアログを開きます。[タイプ] に [Reality Check] を、そのサブタイプに [Constraint] か [Test Input] を選択します。そのうえで方程式を入力します。



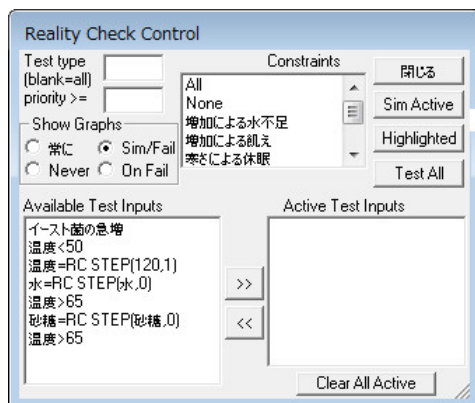
[タイプ] に [Reality Check] を選択すると、通常 [範囲:] となっているダイアログの下部分が、[TypPrio:] に変わります。ここには、Type と Priority を入力します。Priority には 1 から 10 までの優先順位を、Type には 0 から 64 までの値を入れます。

#### 14.5 リアリティチェックの実行

リアリティチェックは、最初のシミュレーションの設定で実行され、次に、入力したリアリティチェック方程式をテストします。

リアリティチェック方程式は、ツールバーもしくは、シミュレーションコントロールダイアログ（詳細は、リファレンスマニュアルの 8 章にあります）から開始できます。他の通常のシミュレーションと同じように、変更や調整ができます。このあと、シミュレーションコントロールダイアログのリアリティチェックをクリックするか、もしくは、ツールバーのリアリティチェックの実行  をクリックします。

リアリティチェック方程式を開始した場合、リアリティチェック制御ダイアログが現れます。



**Test type** (PLE と PLE PLUS は対応しません) は、リアリティチェックを行う方法を特定します。ここが空白ならば、すべての方法がテストされます。リアリティチェックの方法は、方程式編集における [ TypPrio ] において設定します。テキストエディタを使う場合は、単位を設定する場所に [] で閉じて設定されます。例えば (~[type,priority]) となります。この設定は、[ Test All ] を用いる場合のみ有効です。

**Priority>=** (PLE と PLE PLUS は対応しません) は、一定の条件以上という優先順位を持つ制約だけを対象にします。この設定は、[ Test All ] を用いる場合のみ有効です。

**注意** 制約が特定の優先順位を持たない場合は、最大の優先順位を持つものとして扱われます。また制約が特定の方法を持たない場合は、すべての条件を持つものとして扱われます。

**Show Graphs** テストする変数が、本来適合すべき振る舞いに対して、制約を満たしているかを表示します。これは特別のグラフツールを利用し、どの点で間違いが生じているのかを理解するうえで有効です。

- [ 常に ] それぞれの制約条件について、因果グラフを描きます。
- [ Never ] グラフ出力をしません。
- [ Sim/Fail ] リアリティチェックが失敗した際にグラフを表示します。また、[ Sim Active ] や [ Highlighted ] ボタンによって、シミュレーションを可能にします。
- [ On Fail ] リアリティチェックが失敗した時だけグラフを表示します。[ Test All ] を使う場合は、[ Sim/Fail ] と同じことになります。また、[ Sim Active ] や [ Highlighted ] を使う場合は、チェックが失敗するまではグラフ表示されません。

**Constraints** 入力された制約のリストです。リストのなかの 1 つをクリックすることで、それに対応するテスト入力が増強表示されます。こうしたテスト入力を複数選択することもできます。

**Test Inputs** モデルにおけるテスト入力のリストです。このリストは、定義されたテスト入力のすべてです。論理表現による比較は、制約式の要素の状況です。比較は直接行われます。

>> 強調表示されたテスト入力リストを利用可能にします。

<< 強調表示された利用可能なテスト入力を取り除きます。

リストにある要素をクリックすることで強調表示されます。**[Ctrl]**+ クリックは、強調表示の追加・削除となります。さらにダブルクリックは、リスト間の移動となります。

**Active Test Inputs** 利用可能なテスト入力を示します。これらはシミュレーションボタンがクリックされた際に利用されます。クリックすることで要素が強調表示されます。**[Ctrl]**+ クリックは、強調表示のスイッチになっています。ダブルクリックは、リストから除外します。

**Clear All Active** [ Active Test Inputs ] の要素を取り除きます。

**Sim Active** [ Active Test Inputs ] の要素を利用してシミュレーションを行います。それ以外の制約は無視されます。シミュレーションが問題なく実行されると、結果は通常のものと同様に保存されます。その結果は、ツールを使って分析することができます。しかし、因果の探索が、期待通りのものになるとは限りません。なぜなら、シミュレーションの間にモデルの構造が変化している可能性があるためです。

**Highlighted simulates** 制約リストにおいて強調表示されている要素を用いてシミュレーションを行います。制約の論理構造の状況によっては、複数のシミュレーションが必要になることもあります。最後に実施されたシミュレーションは、通常のものと同様に保存されます。

**Test All** 全ての制約をそれぞれテストします。制約の論理構造の状況によっては、複数のシミュレーションが必要になることもあります。このテストは時間を要します。そのエラーなどはウィンドウに表示されます。**[ Test All ]** はシミュレーションの結果を保存しませんが、リアリティチェックの結果を表示します。

**Close** リアリティチェック制御ダイアログを閉じます。

### 14.5.1 リアリティチェックツール

分析ツールセットに、リアリティチェックツールを追加することができます ( PLE と PLE PLUS は対応しません )。このツールはリアリティチェックを行うショートカットであり、制約を強調表示し、**[ Highlighted ]** をクリックします。現在選択されている制約を使います。もし選択されているものが制約でなければエラーメッセージを表示します。

### 14.5.2 リアリティチェックの結果

リアリティチェックの結果は、テキストウィンドウに表示されます。そこには、どの制約がチェックされ、どの制約が破られたのかを示します。


このウィンドウは、**[ Sim Active ]** や **[ Highlighted ]** もしくは **[ Test All ]** が実行されるたびに表示されます。

### 14.5.3 シミュレーション結果の検討

**[ Sim Active ]** もしくは **[ Highlighted ]** をクリックするごとに新しいシミュレーションが作成されます。このシミュレーションは、シミュレーション名の編集ボックスの名前で



作成されます。リアリティチェック制御ダイアログが開いている間は、シミュレーションの結果は通常のものと同様に扱えます。したがって別のシミュレーションを作成したり、別の結果と比較検討することができます。しかしシミュレーションの結果は、上書きされます。もし2種類のリアリティチェックを比較したい場合は、1度リアリティチェック制御ダイアログを閉じて、別のシミュレーション名でリアリティチェックを実行します。

注意 シミュレーションの準備  を用いて、定数の値を変化させた場合は、リアリティチェックを実行中はその値が用いられます。

## 14.6 イースト菌の成長における現実性チェック

リアリティチェック方程式の書き方と使い方の手順を示すには、単純な例を使って操作することが有効です。水の入った皿の中のイースト菌の成長をモデル化するとします。水は、一定の温度で、一定量の砂糖が入った状態です。

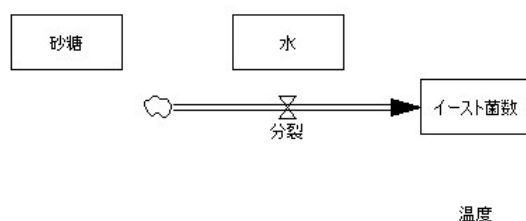
まず始めに、満たされるべきリアリティチェック方程式を、いくつか挙げます。

- 温度が 50 度以下になると、イースト菌の成長は停止し、休眠します。
- 温度が 100 度以上になると、イースト菌は死滅します。
- 砂糖がなく、休眠状態でなければ、イースト菌は死滅します。
- 水がなく、休眠状態でなければ、イースト菌は死滅します。
- イースト菌が成長し続ければ、全ての砂糖を消費します。
- イースト菌が成長し続ければ、全ての水を消費します。

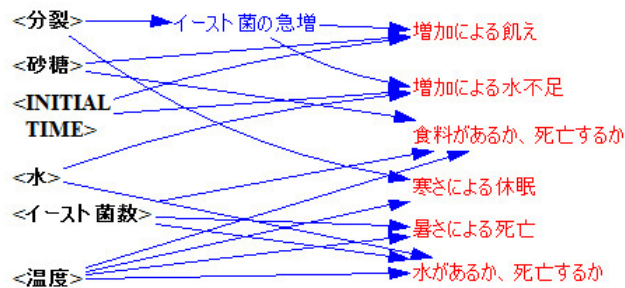
またイースト菌は、分裂により再生し、適切な条件であれば、10 分で倍増します。

### 14.6.1 テスト入力と制約方程式

リアリティチェック方程式のテストが必要な変数を定義することにより、モデルの作成を開始します。そして、リアリティチェック方程式を入力します。リアリティチェック方程式は、イースト菌の数、水の量、砂糖の量、イースト菌の分裂数、温度についての概略を表します。ストックを確認しながら、以下のような骨格のモデルを作成します。



わかりやすくするために、2 つめの表示画面に、リアリティチェック方程式を作成します。まず始めに、モデルのすべての要素を代行変数として作成することです。代行変数の使用により、リアリティチェック図への修正をすることなく、モデル構造を変更することができます。



リアリティチェック図を作成する上で、決められたルールはありません。経験上、通常モデル変数、テスト入力、制約ごとに縦に並べるのが最も理解しやすい形です。また、色で識別することも有効です（例えばテスト入力は青とし、制約を赤にする）。矢印が非常に乱雑になりますが、それはリアリティチェックの情報によって影響をうけます。しかし矢印の乱雑さは、大きな問題ではありません。

方程式編集ダイアログに、制約とテスト入力を入力します。

方程式の [ タイプ ] を [ Reality Check ] とし、サブタイプを [ Constraint ] とします。条件と結果は 2 つのウィンドウに分けられます。

リアリティチェック方程式は、以下のようになります。

寒さによる休眠 : THE CONDITION : 温度 < 50 : IMPLIES :  
分裂 = 0

暑さによる死亡 : THE CONDITION : 温度 = RC STEP(120,1)  
: IMPLIES : イースト菌数 <= RC DECAY CHECK(1, イースト菌数, 2)

「暑さによる死亡」では、「温度」に STEP 関数を使用し、「イースト菌数」をチェックするために、RC DECAY CHECK 関数を使用します。このモデルでは、死滅する条件に

注目していることから、decay【衰退】は[イースト菌数]の振る舞いを見るのに適しています。

```
食料があるか、死亡するか : THE CONDITION :
砂糖 = RC STEP(砂糖,0) : AND : 温度 > 65 : IMPLIES :
イースト菌数 <= RC DECAY CHECK(1, イースト菌数,15)
```

この制約は、平均死亡時間 15 で「イースト菌数」がゼロに減少することです。

```
水があるか、死亡するか : THE CONDITION : 水 = RC STEP(水,0) : AND :
温度 > 65 :
IMPLIES : イースト菌数 <= RC DECAY CHECK(1, イースト菌数,1)
```

```
イースト菌の急増 : TEST INPUT : 分裂 = 1e+022
```

```
増加による飢え : THE CONDITION : イースト菌の急増 : IMPLIES :
砂糖 <= RC DECAY CHECK(1, 砂糖,0.5,INITIAL TIME)
```

```
増加による水不足 : THE CONDITION : イースト菌の急増 : IMPLIES :
水 <= RC DECAY CHECK(1, 水,0.5,INITIAL TIME)
```

最後の 2 つの制約について、テスト入力が、シミュレーションの始めから開始するため、RC DECAY CHECK 関数は、INITIAL TIME でチェックし始めることに注目してください。もしテスト入力が、シミュレーション中に開始するために RC STEP 関数を使用している場合、RC DECAY CHECK 関数を使った最後の 2 つで INITIAL TIME の引数は無視されます。

最後の制約です。

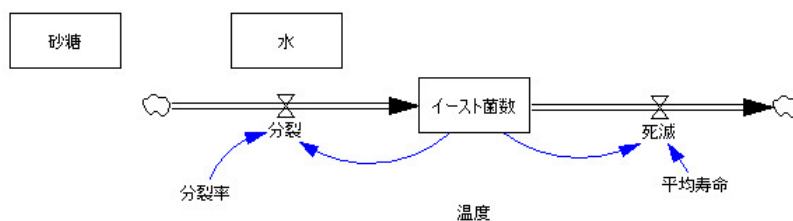
```
RC START TIME = 10
```

時間は任意です。開始が早ければ、イースト菌はわずかで、砂糖が多くなります。一方、開始が遅ければ、イースト菌が多くなり、砂糖が少なくなります。リアリティチェックを無事クリアしたのち、改めて RC START TIME を変更し、再度テストすることは有効な方法です。

### 14.6.2 初期モデル

リアリティチェック方程式の使用を可能にする簡単なモデルがあります。指数的な増加についての基本的な知識があれば、モデルを作成するために骨格を作成することができます。できるだけ簡単なモデルで始めます (*yeast1.mdl*)。

モデルは英語版です。



```
FINAL TIME = 300
```

```
Units : 分
```

```
INITIAL TIME = 0
```

```

Units : 分

イースト菌数 = INTEG( 分裂 - 死滅, 100)
Units : 細胞

分裂 = イースト菌数 * 分裂率
Units : 細胞/分

分裂率 = 0.08
Units : 1/分

平均寿命 = 250
Units : 分

死滅 = イースト菌数 / 平均寿命
Units : 細胞/分



水 = 100
Units : ml

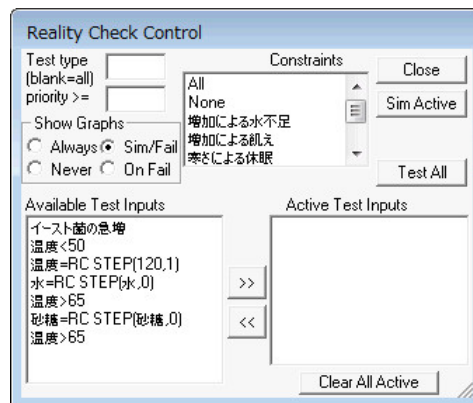
温度 = 85
Units : 華氏度

砂糖 = 100
Units : g

```

このモデルでは、「砂糖」、「水」、「温度」を使用しません。そして、方程式編集ダイアログの [ タイプ ] で [ 定数 ] に設定されています。

次に、このモデルでリアリティチェック方程式を実行します。実現可能性の実行  をクリックするか、もしくはシミュレーションの制御  のダイアログで、[ リアリティチェック ] をクリックします。



リアリティチェックのダイアログで、モデルのすべての制約リストが表示されます。その下は、テスト入力のリストです。他のものが、制約の条件にもとづき作成されているのに対して、「イースト菌の急増」だけは名前で指定されています。

[ Test All ] をクリックすることで、一連のシミュレーションを開始します。以下の制約違反が表示されます。

```

Starting testing of Constraint- 増加による水不足
Test inputs :
  イースト菌の急増
... testing - 増加による水不足
The constraint -増加による水不足 - violated at time 1
-----

Starting testing of Constraint- 増加による飢え
Test inputs :
  イースト菌の急増
... testing - 増加による飢え
The constraint -増加による飢え - violated at time 1
-----

Starting testing of Constraint- 寒さによる休眠
Test inputs :
  温度<50
... testing - 寒さによる休眠
The constraint -寒さによる休眠 - violated at time 0
-----

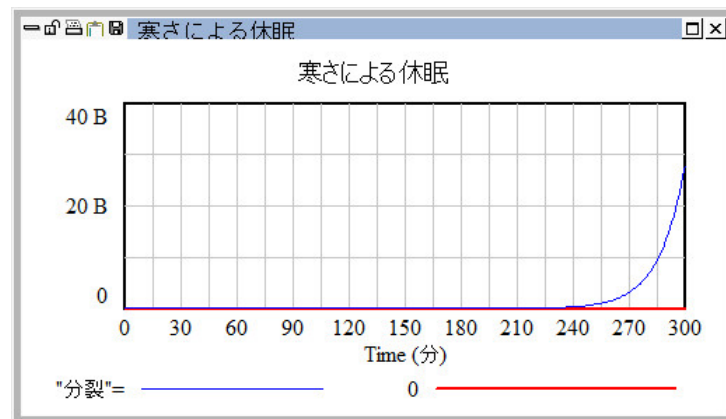
Starting testing of Constraint- 暑さによる死亡
Test inputs :
  温度=RC STEP(120,1)
... testing - 暑さによる死亡
The constraint -暑さによる死亡 - violated at time 11
-----

Starting testing of Constraint- 水があるか、死亡するか
Test inputs :
  水=RC STEP(水,0)
  温度>65
... testing - 水があるか、死亡するか
The constraint -水があるか、死亡するか - violated at time 11
-----

Starting testing of Constraint- 食料があるか、死亡するか
Test inputs :
  砂糖=RC STEP(砂糖,0)
  温度>65
... testing - 食料があるか、死亡するか
The constraint -食料があるか、死亡するか - violated at time 11

```

それぞれの制約に違反があります。制約に含まれた要素がモデルに関連していないので、驚くべきことではありません。示されたモデルは、基本的な成長の仕組みを示しています。すべての制約が関連する制御や抑制には、注意が払われていません。エラーウィンドウに加えて、以下のような6つのグラフが表示されます。



上の線は、分裂を示し、下の線は本来あるべき線を示しています。

レポートウィンドウは結果の要約で終了しています。

```
*****
0 successes and 6 failures testing 6 リアリティチェック equations
The リアリティチェック Index as run is 0
Closeness score is 0.0 % on 6 measurements
```

初めの行は、要約数です。二番目の行は、リアリティチェックのインデックスを示しています。これは、モデルにおけるダイナミックな変数によって分けられた連続区分を示しています。すべての変数に対して、1つかそれ以上のリアリティチェック方程式の可能性があるので、このインデックスは、完全なリアリティチェックセットを持ったモデルに近いものです。

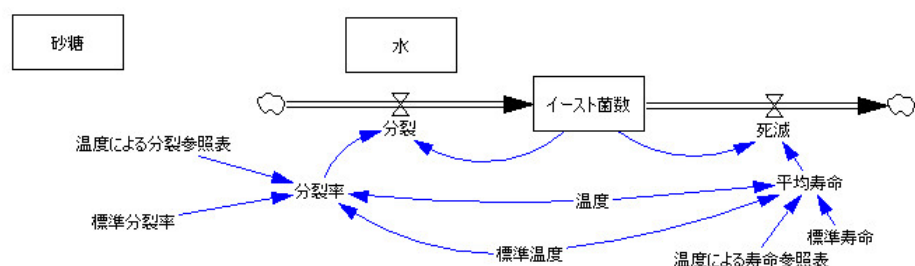
最後に、近似スコアが示されます。近似スコアは、リアリティチェックの平均近似値です。もし、リアリティチェックがすべて守られれば、その近似スコアは1です。もしパスしなければ、1からパスしなかった制約を全制約で割ったものを差し引きします。

このように、リアリティチェックが失敗すれば近似スコアは0となり、制約が破られた程度を示す指標となります。

### 14.6.3 温度と分裂そして終結

成長温度に関する2つの状況があります。温度が低ければ、すべて休眠します。温度が高ければ、イースト菌は死亡します。これを念頭において、「分裂」と「死滅」に、簡単な方程式を設定します。以下のように *yeast2.mdl* を作成します。

モデルは英語版です。



分裂 = イースト菌数 \* 分裂率

分裂率 = 標準分裂率 \* 温度による分裂参照表 (温度/標準温度)

標準分裂率 = 0.08

標準温度 = 80

温度による分裂参照表 ((0,0),(0.8,0),(1,1),(2,1))

死滅 = イースト菌数 / 平均寿命

平均寿命 = 標準寿命 \* 温度による寿命参照表 (温度/標準温度)

標準寿命 = 250

温度による寿命参照表 ((0,1),(1,1),(1.25,0.02),(2,0.001))

制約をテストすると、「寒さによる休眠」という制約が破られています。

イースト菌は、温度が高くなってもすぐには死滅しないため、制約は破られます。この場合、どちらがより現実的かを考えて、制約を満たすモデルを再度作成するか、もしくは制約を少し緩めます。制約を緩める 1 つの方法は、イースト菌が死滅するまでの時間をより長くするために、時間の経過を変更することです (RC DECAY CHECK の引数を 2 から 5 に変更します)。

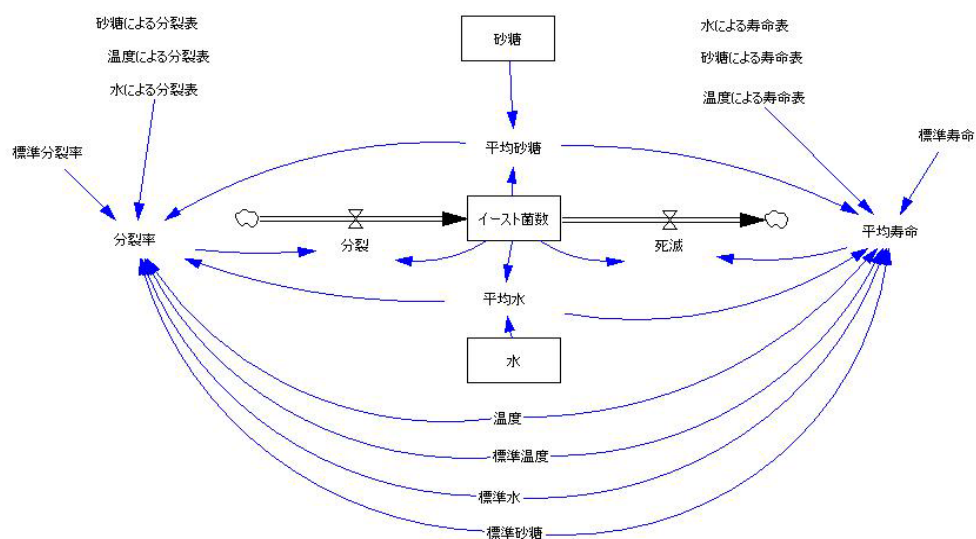
```
「暑さによる死亡」 : THE CONDITION : 温度 = RC STEP(120,1)
: IMPLIES : イースト菌数 <= RC DECAY CHECK(1, イースト菌数,5)
```

制約をテストすると、「寒さによる休眠」と「暑さによる死亡」のどちらも守られていることがわかります。制約の厳密さと、方程式の修正をやり取りするこの種の方法は有効なものです。それは、些細なことで破られる制約に、より注意を払うことができるためです。

#### 14.6.4 水と砂糖から影響を受ける分裂

次の 2 つの制約方程式は、生存するには水と砂糖が必要であることを示しています。イースト菌の分裂率と、イースト菌の平均寿命に対して、水と砂糖の不足が及ぼす影響をモデルにします。新しいモデル、*yeast3.mdl* は以下のようになります。

英語版です。



方程式はサンプルモデルにあるものを利用できます。テスト入力は、「水」と「砂糖」が存在する極端な状況が想定され、「死滅」が急増するものです。オイラ積分を利用するため、「イースト菌数」が減少に向かうと、振幅を繰り返します。これを防ぐには、別の積分法を利用するか、もしくは「死滅」の方程式を以下のように変更します。

$$\text{死滅} = \text{MIN}(\text{イースト菌数} / \text{TIME STEP}, \text{イースト菌数} / \text{平均寿命})$$

この公式は、退屈なダイナミックスの発生を防ぐだけでなく、「イースト菌数」を確実にゼロに設定することができます。そのためには、「イースト菌数」がゼロの時、「平均砂糖」と「平均水」において、計算上のエラー（オーバーフロー）を防ぐために、以下のような方程式が必要になります。

$$\begin{aligned} \text{平均砂糖} &= \text{ZIDZ}(\text{砂糖}, \text{イースト菌数}) \\ \text{平均水} &= \text{ZIDZ}(\text{水}, \text{イースト菌数}) \end{aligned}$$

構造の変更により、モデルは 6 つの制約のうち、4 つをパスしました。

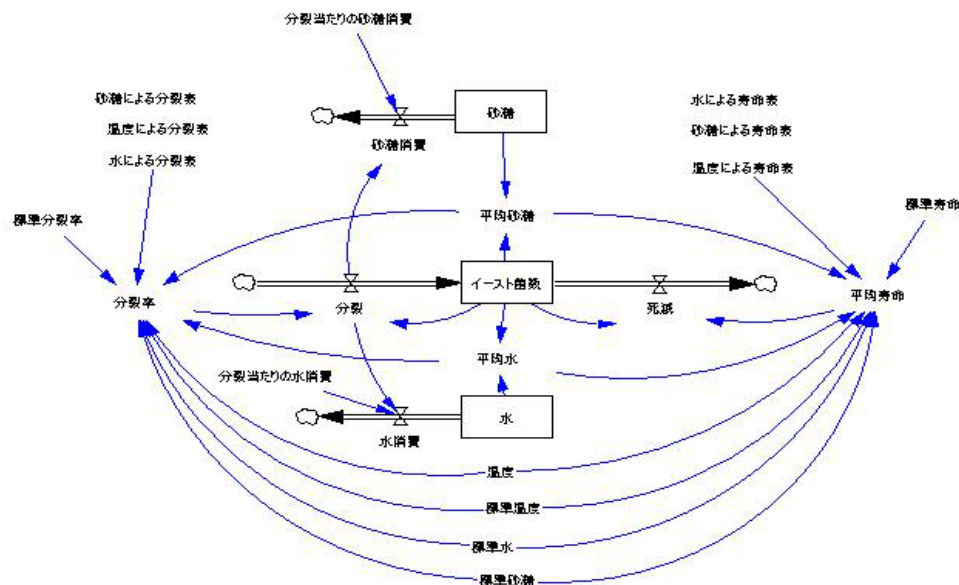
#### 14.6.5 分裂により影響を受ける水と砂糖

完成に近づいていますが、最後の 2 つの制約が残っています。残されている問題は、イースト菌の増加が水と砂糖の変化に関連付けられていないということです。これをモデル化する簡単な方法は、分裂の過程を、砂糖と水の消費として扱うことです。

英語版です。

*yeast4.mdl* は以下ようになります。





このモデルは、書かれたすべての制約をパスします。しかし、これは良いモデルでしょうか。その答えはおそらく“いいえ”です。イースト菌の成長の正確な表現には、実証データが必要です。また、温度や砂糖の不足に対するモデルの調整も必要です。このモデルは、作成したリアリティチェック方程式に反していないだけです。

いくつかのモデルを見てきました。リアリティチェック方程式とはどういうものでしょう。「暑さによる死亡」の小さな変化を除いて、変化はなく、全く同じ図が使われました。

#### 14.6.6 結論

リアリティチェックを使用することの目的は2つあります。第1に、モデルが理にかなうために必要な事項を作成します。これらは、モデル作成において最も重要な事項ですが、あまり文章化されることがありませんでした。イースト菌が成長を続けることで、結局水を使い果たすというような非常に単純な認識は、システムを理解するうえで重要なことです。

第2に、モデル作成の方法を図示することは、効果的で効率がよいということです。モデル作成を基本的な中心部分から始め、制約に対応した補完的な構造を追加します。これは、モデル作成の厳正さと方向を決め、結果的にモデル作成の速度と質を改善します。



## 第 15 章

# 感度テスト

感度テストには、Vensim PLE は対応しません。

### 15.1 モンテカルロシミュレーション

感度テストは、定数の値についての仮定を変更し、その出力結果を検討します。手動による感度テストは、定数（場合によっては複数の定数）の値を変更し、シミュレーションを何度も繰り返し、出力される値の広がりを見ます。

モンテカルロシミュレーションは、multivariate sensitivity simulation (MVSS)【多変量の感度シミュレーション】としても知られており、この処理を自動的に行います。定数の設定幅の中から抽出された値で、百あるいは千のシミュレーションを実行し、出力結果が保存されます。ラテンハイパーキューブサンプリングは、大きなモデルで感度テストを早く行うための特別なものです。

### 15.2 市場成長モデル (sales.mdl)

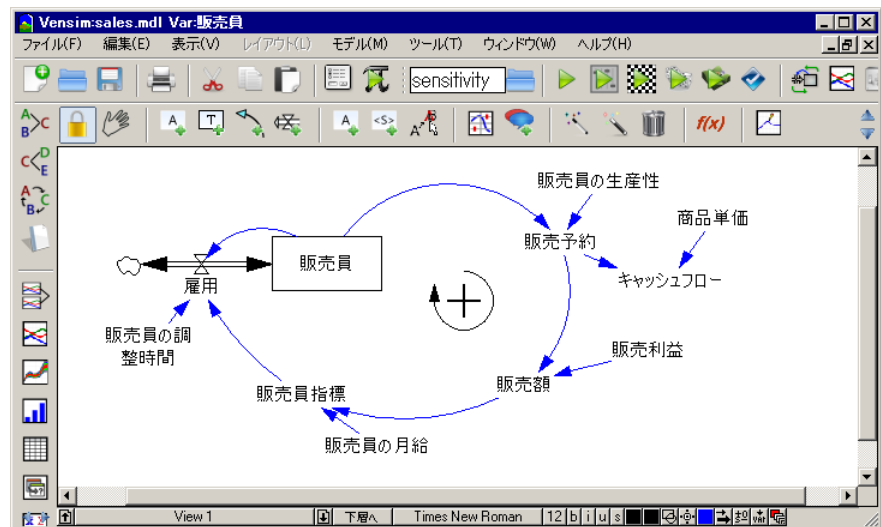
正のフィードバックループを持つ販売モデルを以下に示します。より多くの販売員がより多くの売り上げをもたらし、それによって収入は増加し、より多くの販売員が雇用されます。負のフィードバックループは、雇用と解雇を通じた販売の調整があります。

➤ `guide\chap15\sales.mdl` を開きます。

英語版です

もしくは、以下の方法をとります。

➤ 以下のような図と方程式のモデルを作成し、`guide\chap15` に、異なる名前で保存します（例えば `sales1.mdl`）。[ 時間の範囲 ] は、[ 開始時間 ] = 「0」、[ 終了時間 ] = 「60」、[ 時間ステップ ] = 「0.25」とし、[ 時間単位 ] は「月」とします。



### 15.2.1 Sales.mdl の方程式

キャッシュフロー = 販売予約 \* 商品単価  
Units : 円/月

販売員指標 = 販売額 / 販売員の月給  
Units : 人

雇用 = (販売員指標 - 販売員) / 販売員の調整時間  
Units : 人/月

販売予約 = 販売員 \* 販売員の生産性  
Units : unit/月

商品単価 = 100  
Units : 円/unit

販売利益 = 10  
Units : 円/unit

販売額 = 販売予約 \* 販売利益  
Units : 円/月

販売員 = INTEG (雇用, 50)  
Units : 人

販売員の調整時間 = 6  
Units : 月

販売員の生産性 = 210  
Units : unit/(人 \* 月)




販売員の月給 = 2000  
Units : 円/(人 \* 月)

### 15.2.2 初期設定のシミュレーション

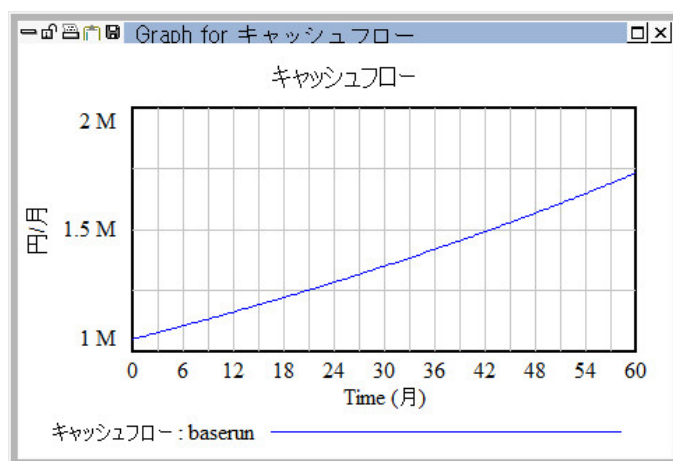
モデルの初期設定での定数を用いて、シミュレーションの振る舞いを見ます。


➤ シミュレーション名の編集ボックス  をダブルクリックし、

「baserun」と入力します。

- シミュレーションの実行  をクリックします。
- 「販売員」をクリックし、グラフ  をクリックします。「キャッシュフロー」をクリックし、グラフ  をクリックします。

「販売員」と「キャッシュフロー」の両方が、ゆっくり増大しています。




- 制御パネル  の [ データセット ] タブを選択し、[ baserun ] をダブルクリックし、取り除きます。


## 15.3 複数のパラメータによる不確実性


このモデルには、シミュレーション出力の効果を検証できる 5 つの定数があります。ここでは、「商品単価」と「販売利益」という定数の値は正確だとみなします（これらは、経営者が設定できる事項だからです）。不確定なパラメータは、「販売員の生産性」、「販売員の調整時間」、「販売員の月給」です。これらの定数の変化が、モデルの振る舞いに与える影響を見るために、定数の値を無作為に割り当てます。1 つのパラメータに対する感度分析を行うには、1 つのパラメータを選択します。

**注意** パラメータは、定数と同義語です。パラメータは、感度分析と最適化において、変化させる定数を示すものとしてよく用いられます。

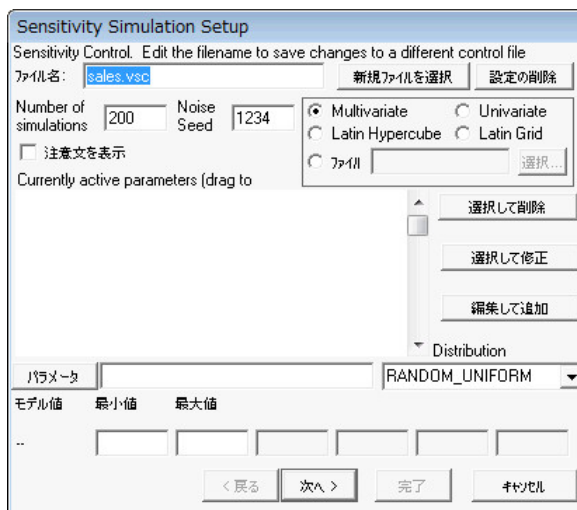
### 15.3.1 感度分析の管理パラメータ

Vensim には、感度シミュレーションの設定方法が 2 つあります。1 つは、シミュレーションコントロールの [ Sensitivity ] タブを使用することです。もう 1 つは、ツールバーにある感度分析シミュレーションの開始  をクリックし、感度分析ウィザードを使用することです。

- シミュレーション名の編集ボックス  をダブルクリックし、「sensitivity」と入力します（baserun となっています）。

➤ ツールバーの、感度分析シミュレーションの開始  をクリックします。

感度分析の管理ダイアログが開きます。



➤ [ Multivariate ]【多変量】のラジオボタンが選択されていることと、[ Number of Simulations ]【シミュレーション回数】が「200」に設定されていることを確認してください。

モンテカルロ多変量感度分析は、与えられた範囲内の値を抽出して実行されます。1つのテストを実行するために、その分布が設定されたパラメータから抽出され、その値がシミュレーションで利用されます。[ Number of Simulations ]【シミュレーションの回数】が [ 200 ] に設定されている場合、この過程は 200 回繰り返されます。

➤ [ パラメータ ] ボタンをクリックすると、パラメータダイアログが開き、感度分析で選択できる、すべてのパラメータ（定数）が、表示されます。「販売員の生産性」をクリックし、[ OK ] をクリックします。

### 15.3.2 無作為の一様分布

感度シミュレーションを実行するには、各パラメータに、確率分布を設定する必要があります。最も簡単な分布は、無作為の一様分布です。これは、最大値と最小値の間にある値が、均等に生じます。無作為の一様分布は、ほとんどの感度テストに適しており、デフォルトで選択されています。その他の一般的に使用される分布は正規分布で、平均に近い値ほど生じやすい特徴があります。

Vensim は、いくつかの分布を選択することができます。最も一般的に利用される分布は、一様分布、正規分布、三角分布です。もし、特別な分布を選択する理由がなければ、一様分布を用います。

最小値と最大値が各パラメータの値がとる範囲になります。[ パラメータ ] ボタンの下に、[ モデル値 ] が [ 210 ] と表示されています。

➤ [ 最小値 ] をクリックし、「200」とします。[ 最大値 ] を「220」とします。

最小値の 200 は、販売員が達成する最低の生産性を表し、最大値の 200 は、最高の生産性を表しています。

➤ [ 編集して追加 ] をクリックします。

➤ [ パラメータ ] をクリックし、「販売員の調整時間」をクリックし、[ OK ] をクリックします。

➤ [ 最小値 ] を「3」とし、[ 最大値 ] を「12」とします。これらの値はモデルが持つ初期値「6」に対して非対称です。なぜなら値が、モデル値より少し小さいか、かなり大きいと考えたためです。[ 編集して追加 ] をクリックします。

➤ [ パラメータ ] をクリックし、「販売員の月給」をクリックし、[ OK ] をクリックします。

### 15.3.3 無作為の正規分布

「販売員の月給」で、デフォルトとは異なる分布を選択します。[ RANDOM NORMAL ]【正規分布】は正規分布に従った値を抽出します。その設定には、平均と標準偏差に加えて、最大値と最小値の範囲を指定します（技術的には、これは切断正規分布と呼ばれます。これに対して通常の正規分布は範囲に限界がなく、非常に大きな正や負の値が生じます）。

➤ [ Distribution ]【分布】のドロップダウンボックス[ ] から、[ RANDOM NORMAL ] を選択します（スクロールバーで上に移動します）。

感度分析の管理ダイアログの下方に、編集ボックスが追加されます。

➤ [ 最小値 ] を [ 1800 ] とし、[ 最大値 ] を [ 2200 ] とします。[ Mean ]【平均値】を [ 2000 ] とし、[ Standard Deviation ]【標準偏差】を [ 100 ] とします。[ 編集して追加 ] をクリックします。

感度分析の管理ダイアログは以下のようになります。

**Sensitivity Simulation Setup**

Sensitivity Control. Edit the filename to save changes to a different control file  
 ファイル名: sales.vsc [新規ファイルを選択] [設定の削除]

Number of simulations: 200 Noise Seed: 1234  
☒ Multivariate ☐ Univariate  
☐ Latin Hypercube ☐ Latin Grid  
☐ ファイル [選択...]

☐ 注意文を表示

Currently active parameters (drag to  
 販売員の生産性=RANDOM\_UNIFORM(200,220)  
 販売員の調整時間=RANDOM\_UNIFORM(3,12)

[選択して削除] [選択して修正] [編集して追加]

パラメータ: 販売員の月給 Distribution: RANDOM\_NORMAL

モデル値	最小値	最大値	Mean	Standard Deviation
2000	1800	2200	2000	100

[戻る] [次へ>] [完了] [キャンセル]

➤ [次へ] をクリックし、保存リストダイアログを開きます。

## 15.4 保存リスト

保存リストは、保存すべき変数を設定するファイルです。感度シミュレーションは、莫大な量のデータを生成します。したがって、本当に重要な変数データだけを保存するようにします。モデルの全ての変数を感度分析で保存すると、長い時間と、大きなディスク容量を必要とします。

**Sensitivity Simulation Setup**

Savelist Control. Edit the filename to save changes to a different control file  
 ファイル名: sales.lst [新規ファイルを選択] [設定の削除]

List of Variables to be Saved (drag to reorder)

[選択して削除] [選択して修正] [編集して追加]

[選択]

For subscripted variables leave the subscripts off to save all elements.

[戻る] [次へ>] [完了] [キャンセル]

➤ [選択] をクリックし、保存する変数のリストが開きます。[キャッシュフロー] を選択し、[OK] をクリックします。[編集して追加] をクリックします。

➤ [選択] をクリックし、[販売員] を選択し、[OK] をクリックします。[編集して追加] をクリックします。

注意 これらの名前を、編集ボックスの中に個別に入力し、[編集して追加] をクリック




することもできます。

## 15.5 感度分析シミュレーション

➤ [完了] をクリックします。

モデルが一度シミュレーションされ、その後、「販売員の生産性」、「販売員の調整時間」、「販売員の月給」の値が自動的に変更しながら、200 回の追加のシミュレーションが実行されます。データセットは、全ての変数のデフォルトでの振る舞いに加えて、200 回の感度分析シミュレーションによって生成された「キャッシュフロー」と「販売員」の値の範囲を持ちます。


➤ 制御パネル  をクリックし、[データセット] タブをクリックします。[ baserun ] をダブルクリックし、削除します。

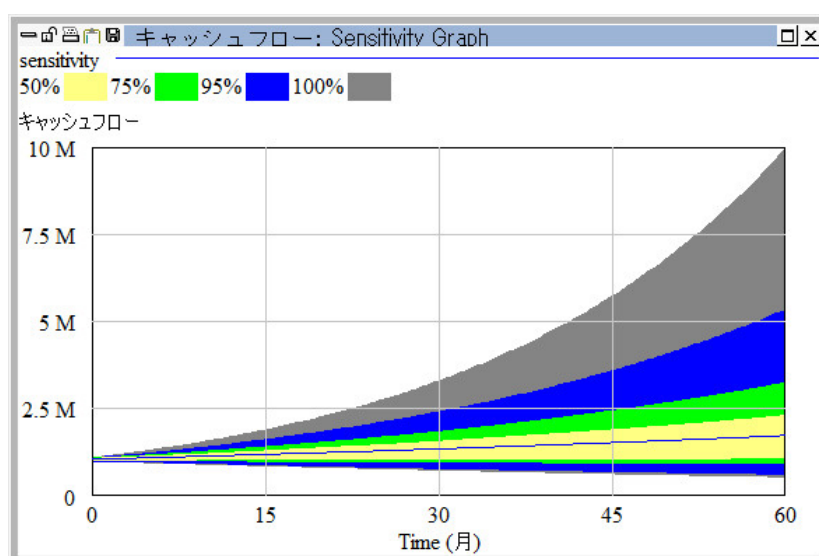
➤ スケッチ上で、「キャッシュフロー」をクリックし、ワークベンチ変数とします。

## 15.6 感度分析の結果出力

感度テストの結果は、いくつかの異なる形式で表示されます。時間グラフは、変数の時間に伴う振る舞いを表示します。変数のとる値の幅は、信頼幅やシミュレーションの個別結果で示されます。

### 15.6.1 信頼幅

➤ 感度グラフ  をクリックします。このデフォルトの形式は、信頼幅を描きます。





グラフは、3 つのパラメータがその分布からランダムな値を取った際に、「キャッシュフロー」がとる値の信頼幅を示します。右上の閉じるボタンの左にある最大化ボタンをク

リックすることで、グラフを全画面表示することができます。

一番外側の範囲（100%）は、シミュレーションの最後に、およそ 1 千万円の最大値と、50 万円の最小値の幅を持っています。「キャッシュフロー」が減少する可能性に注意してください。最初の（モデル初期値での）シミュレーションは、「sensitivity」で示された青いラインとして描かれています。

### 15.6.2 平均値（Vensim PLE Plus は対応しません）


平均値は、信頼幅の間にあり、次のように描かれます。

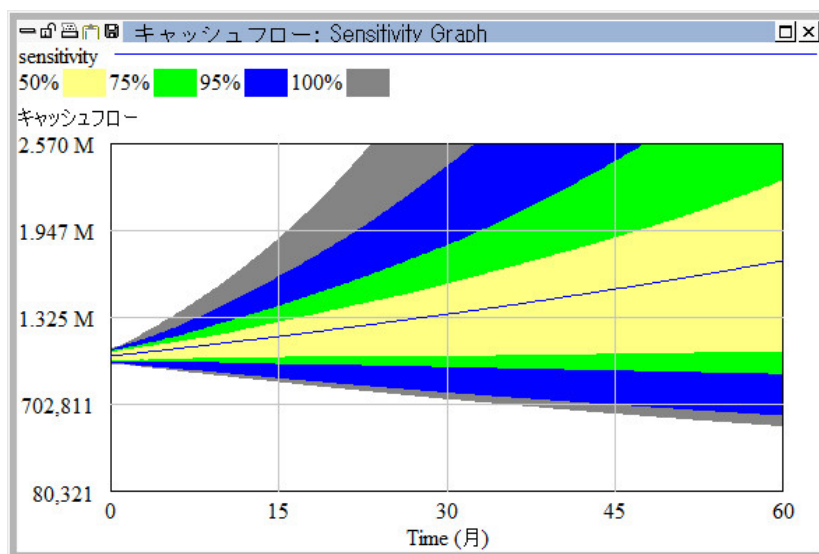
- 感度グラフ  を右クリックします。[ Show Sensitivities as ] の [ Plot Mean Value ] 【平均値を描く】のチェックボックスをクリックします。[ 色 ] で赤を選びます。
- [ Suppress first run plot ] 【初期値での実行結果を描かない】のチェックボックスをクリックします（これで、平均値と信頼幅だけが表示されます）。[ OK ] をクリックします。
- 感度グラフ  をクリックします。

信頼幅の平均値が赤い線で表示されたグラフが、作成されます。


### 15.6.3 グラフ目盛りの拡大



縦軸の値を詳しく見ます。そのために、信頼幅の下限を表示します。

- 「キャッシュフロー」のラインが 2.5M を示すところに、ポインタを置きます。  
[Ctrl] を押しながらマウスを押し、そのままグラフの下までドラッグします（「キャッシュフロー」のラインが 0 を示す部分です）。マウスを放します。
- 感度グラフ  をクリックします。

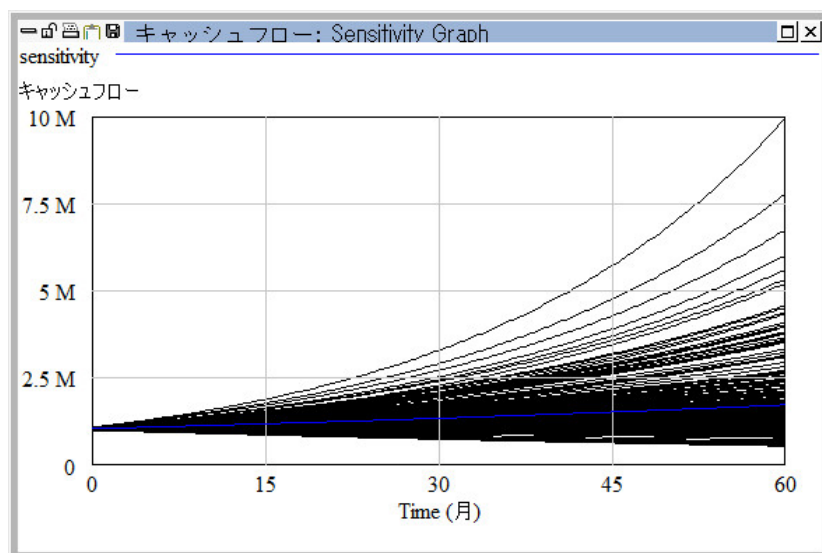



### 15.6.4 個別シミュレーションの表示

感度分析を表示する方法には、シミュレーションの結果を個々に表示することもできます。Vensim PLE Plus では、下の感度グラフ  をクリックします。その他のラインアップでは、以下の方法をとります。

- 感度グラフ  を右クリックします。[ Show Sensitivities as ] の [ Individual Traces ] 【個別シミュレーションの表示】のラジオボタンをクリックし、[ OK ] をクリックします。
- 感度グラフ  をクリックします。


各シミュレーションの個々の結果が描かれます。いくつかの個別の結果を理解するために、このグラフを全画面表示にすることもできます。



- 「販売員」をクリックし、感度グラフ  をクリックします。


非常によく似た振る舞いが見られます。なぜなら、「販売員」のフィードバックループが、「キャッシュフロー」に影響するためです。


## 15.7 感度分析のヒストグラム表示

感度分析の結果は、ヒストグラムとして表示することができます。これはある時点での横断的な値を示します。ヒストグラムは、ある時点でのある範囲にあるシミュレーションの数を表示します。ヒストグラムは、値の分布をみる手段を提供します。Vensim PLE Plus では、棒グラフ  はある時点での感度分析結果を表示するために用いられます。他のラインアップでは、設定を調整します。

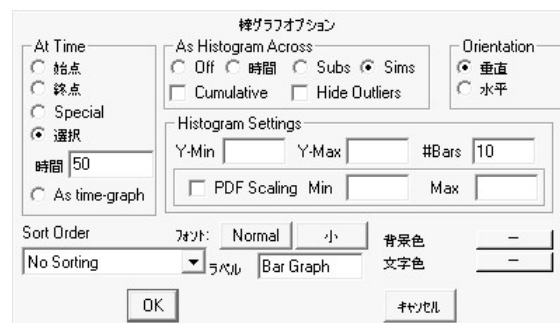
### 15.7.1 デフォルトの分析ツールセットの変更



Vensim には、2 つのデフォルトの分析ツールセットがあります。 *default.vts* は、これまで使用してきたものです。 *default2.vts* は、 *default.vts* より多くのツールを持っています。自分自身でツールを作成し、保存することも可能です。

Vensim PLE Plus では、棒グラフ  をクリックするだけです。他のラインアップでは、以下のようにになります。

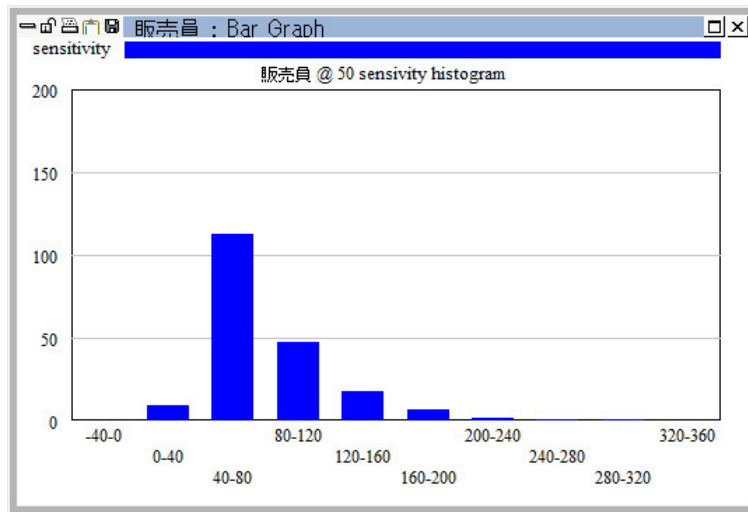
- > [ ツール ] メニュー [ 分析ツール ] [ 開く ] を選択します。
- > [ Do you want to save the Analysis Toolset ] 【分析ツールセットを保存しますか】という表示が出たら、[ いいえ ] をクリックします。
- > *default2.vts* を選択し、[ 開く ] をクリックします。
- > 棒グラフ  を右クリックします。
- > [ As Histogram Across ] の [ Sims ] を選択します。[ At Time ] の [ 選択 ] のラジオボタンをクリックし、[ 時間 ] に「50」と入力します。[ OK ] をクリックします。

棒グラフオプションのダイアログは、以下のようにになります。




- > 制御パネル  の [ データセット ] タブを選択し、「sensitivity」が読み込まれていることを確認します。「販売員」がワークベンチ変数になっているか確認します。
- > 棒グラフ  をクリックします。

感度分析のヒストグラムが、シミュレーションの途中である 50 ヶ月の時点で表示されます。



「販売員」の数は、X 軸に沿って表示されます。Y 軸はシミュレーションの数を表します。つまり、50 ヶ月の時点で、40 から 80 人の「販売員」となるシミュレーションは 115 回あり、80 から 120 人となるのは 50 回です。

### 15.7.2 統計ツール

Vensim Professional もしくは DSS を使用している場合、Statistics  を使用して感度シミュレーションの情報を得ることもできます。このツールについての詳細は、リファレンスマニュアルの 14 章にあります。



## 第 16 章

# モデルにおけるデータ利用

Vensim PLE は表計算ソフトへの直接の接続をサポートしていませんが、データの読み込みを通じた利用は可能です。この方法は、20 章で示している参照モードで利用されます。

### 16.1 データ利用のタイプ

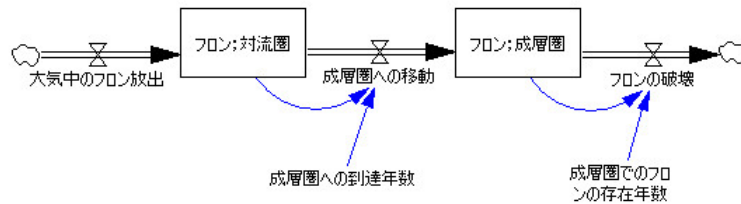
Vensim はデータを 2 通りの方法で使います。第 1 に、モデルを操作する外部入力として、第 2 に、モデルの振る舞いを実際のデータと比較するために利用します。

外部入力はデータ変数であり、モデルの一部を操作する時系列データです。データ変数はシミュレーション中に計算されません。その代わりに、シミュレーション中に既存の時系列データとして利用されます。この時系列データは、データ変数によりデータセットとして読み込まれるか、または、Microsoft Excel や Lotus 123 といった表計算ソフトの値を読み込むデータ関数を持つデータ変数から作成されます。データを利用する第 2 の方法は、Vensim データセットとして実際のデータを読み込み、分析ツールを使用して、モデルの振る舞いを比較することです。本章において表計算ソフトを利用する部分には、コンピュータに Microsoft Excel もしくは Lotus 123 がインストールされている必要があります。

### 16.2 データを使ったモデル操作 (cfc.mdl)

- `guide\chap16\cfc.mdl` を開きます。
- もしくは、以下のような図と方程式のモデルを作成します。[ 時間の範囲 ] は、[ 開始時間 ] = 「1930」、[ 終了時間 ] = 「2130」、[ 時間ステップ ] = 「0.5」とし、[ 時間単位 ] は「年」とします。

英語版です。



### 16.2.1 cfc.mdl 方程式

フロンの破壊 = フロン;成層圏 / 成層圏でのフロンの存在年数  
Units : Mkg/年

フロン;対流圏 = INTEG(大気中のフロン放出-成層圏への移動,0)  
Units : Mkg

フロン;成層圏 = INTEG(成層圏への移動-フロンの破壊,0)  
Units : Mkg

大気中のフロン放出 = A FUNCTION OF(大気中のフロン放出)  
Units : \*\*undefined\*\*

成層圏でのフロンの存在年数 = 55  
Units : 年

成層圏への到達年数 = 5  
Units : 年

成層圏への移動 = フロン;対流圏 / 成層圏への到達年数  
Units : Mkg/年

「大気中のフロン放出」を除いて、このモデルは完成しています。

➤ *guide\chap16* に、異なる名前（例えば *mycfc.mdl*）で保存します。

## 16.3 表計算ソフトからのデータ読み込み（PLE は対応しません）

モデルにデータを取り込む簡単な方法は、Microsoft Excel もしくは Lotus 123 から直接データを読み込む機能を使うことです。この方法は、表計算ソフトでアクセスできるのであれば、どんなデータソースにも利用可能です。

方程式 **f(x)** を選択し、「大気中のフロン放出」をクリックします。

➤ [タイプ] のドロップダウンボックス [ ] から [データ] を選択します。その下のドロップダウンボックス [ ] ([Normal] になっています) から [方程式] を選択します。

➤ [関数] タブをクリックし、[関数クラス] のドロップダウンボックス [ ] から [Data Only] を選択します。Microsoft Excel を使用している場合は、関数のリストから、[GET XLS DATA] を選択します（Lotus 123 を使用している場合は、



[ GET 123 DATA ] を選択します。

以下の 4 つの引数を持つ関数が入力されます。

'filename' 【ファイル名】

'tabname' 【シート名】

'time\_row\_or\_col' 【時間軸が行か列】

'first\_data\_cell' 【最初の時間セル位置】

注意 これらの項目は、シングルクォート ( ' ) で囲んで入力します。これらの入力に、String 変数を使用することも可能ですが、直接文字通り入力の方が容易です。


フロン放出は *guide\chap16* に、実測値と予測値が時系列データとしてあります。

- *cfc11jap.xls* ( Microsoft Excel )
- *cfc11jap.wk4* ( Lotus 123 表計算 )


- Microsoft Excel を使用する場合は、最初の項に、ファイル名 ' ' *cfc11jap.xls* ' ' を入力します。シングルクォート ( ' ) を忘れないでください。Lotus 123 の場合は、' *cfc11jap.wk4* ' と入力します。
- 2 つめの 'tabname' をダブルクリックし、'cfc11jap' とします。これはシート名です。
- 3 つめの 'time\_row\_or\_col' をダブルクリックし、'2' とします。2 は時間が行方向であることを意味します。
- 4 つめの 'first\_data\_cell' をダブルクリックし、'C4' とします。時間が始まる最初のセル位置です。
- [ 単位 ] に 'Mkg/年' を選択し、[ OK ] をクリックし、方程式編集ダイアログを閉じます。

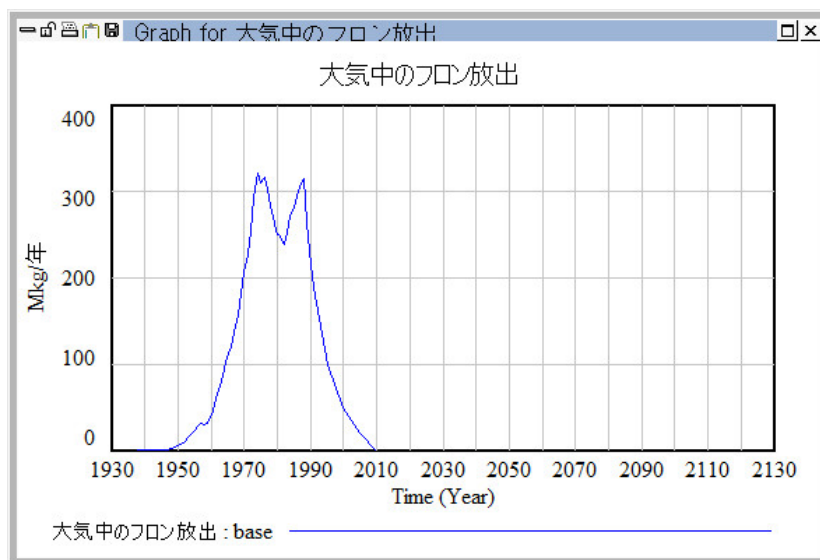
### 16.3.1 シミュレーション

Microsoft Excel 及び Lotus 123 の新しいバージョンでは、Vensim の GET DATA 関数から呼び出され、自動的にスタートしますが、古いバージョンでは、シミュレーションする前にスタートしておく必要があるかもしれません。

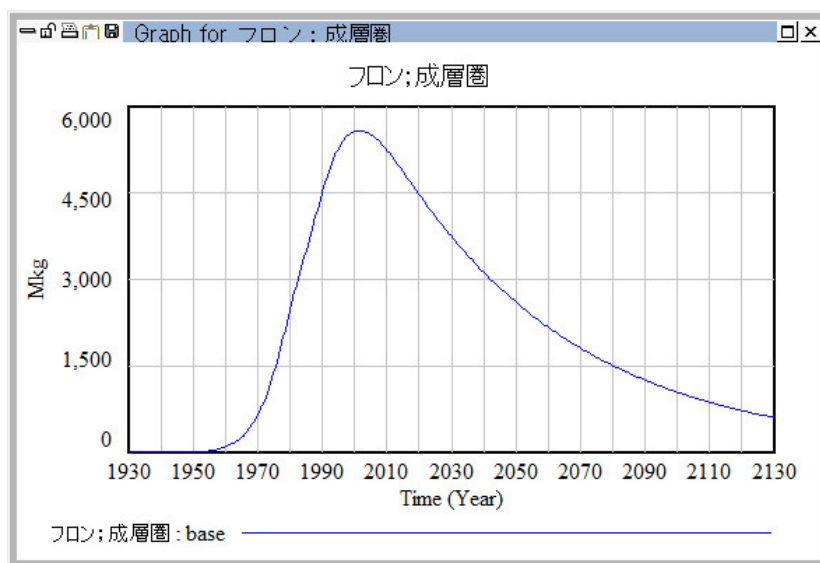
- 実行名を入力し (例えば *base*)、シミュレーションの実行  をクリックします。

Vensim は、自動的に表計算ソフトを開き、*cfc11jap* データから値を読み取るとともに、シミュレーションします。モデルは、フロン放出が発生する 8 年前の、1930 年にシミュレーションを開始します。この時、Vensim はエラーを表示します。なぜなら、モデルが読み込むデータの時間よりも早くシミュレーションが開始するためです。1938 年以前の時系列データに、ゼロを入力することもできますが、元となったデータを直接利用しました。

- 「大気中のフロン放出」をクリックし、ワークベンチ変数とします。グラフ  をクリックします。



- 同じように、「フロン；成層圏」のグラフを作成します。これらのグラフを比較すると、100 年後にも成層圏中に著しいフロンが残存する結果となる長期の遅れに気づきます。



このシュミレーションは、「大気中のフロン放出」に関して、楽観的な予測をしています。さらにこれがもたらす結果について、検討します。

#### 16.4 読み込まれたデータ変数

データを得るために表計算のデータを読み込むのではなく、データセットから値を得ることもできます。このようなデータ変数は、汎用的な subtype を “Normal” にします。データセットは、シミュレーションの実行結果もしくは、他のソースからの読み込みに

よって作成されます。このデータセットは、モデルがシミュレーションされる前に、作成されるか読み込まれている必要があります。

- *cfc.mdl* を、他の名前（例えば *cfc2.mdl*）で保存します。
- 方程式 *f(x)* を選択し、「大気中のフロン放出」をクリックします。[タイプ]の下のドロップダウンボックス[ ]([方程式]になっています)から[Normal]を選択します。[OK]をクリックします。

シミュレーションを実行した時、Vensim はデータ変数を持つ読み込まれたデータセットを捜します。もしデータセットが見つからなかった場合、シミュレーションは停止します。

フロン放出は *guide\chap16* に、実測値と予測値が時系列データとしてあります。実測値は、次の2つのファイルがあります。

- *cfc11jap.dat* 形式（テキスト形式での Vensim データ形成）
- *cfc11jap.tab* 形式（タブ区切りのテキストファイル）

表計算のデータとは異なり、これらのファイルは予測値を含みません。Vensim にデータセットを作成するために、*cfc11jap.dat* のデータファイルを使用します。

- Vensim Professional または DSS の場合、[ファイル]メニュー [ファイル編集] を選択し、*cfc11jap.dat* を選んで [開く] をクリックします。  
もしくは以下のようにします。
- テキストエディタまたはワードプロセッサを用いて、アスキーのテキストファイル *cfc11jap.dat* を開きます。

ファイルを開くのは特に必要な作業ではなく、データを表示することが目的です。具体的には、変数の名前に続き、時間の列、変数の列が続きます。下記の形式でデータが表示されます。

```

大気中のフロン放出↓
1938      0.1      (Mkg)↓
1939      0.1↓
1940      0.1↓
1941      0.1      *****
1942      0.1      *** Source: Alternat
1943      0.2      ***      Washingt
1944      0.2      *** From: Trends '93
1945      0.3      *** D.P. Kaiser, R.J
1946      0.6      *** Information Anal
1947      1.3      *** Oak Ridge, Tenn.
1948      2.3      *****
1949      3.8↓
1950      5.5↓
1951      7.6↓
1952     11.0↓
1953     15.0↓
1954     19.8↓

```

- *cfc11jap.dat* を閉じます。

元あるファイルは、*cfc11jap.dat* と *cfc11jap.tab* で英語版です。

元の英語版ファイルの変数名だけを日本語に変えています。



## 16.5 テキスト形成のデータ読み込み (.dat)

テキストファイルの *cfc11jap.dat* をバイナリ形式の Vensim データファイル *cfc11jap.vdf* に変換し、時系列データとして読み込みます。

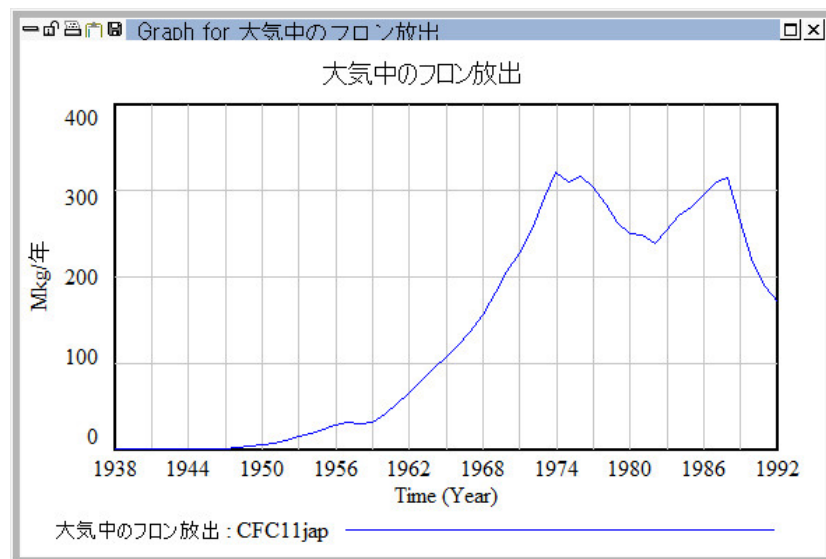
- [モデル]メニュー [ .dat フォーマット読み ] を選択し、[ *cfc11jap.dat* ] を選択し、[ 開く ] をクリックします。

[ Conversion complete without error. ]【エラーなしに変換しました】というメッセージが表示され、読み込まれた変数の数が表示されます。データセット *cfc11jap.vdf* が最初のデータセットとして読み込まれます。分析ツールがこのデータセットに対して稼働します。

日本語を含むデータを読み込む場合は、コード情報として [ Japanese(ShiftJIS) ] を選択する必要があります。

- [ OK ] をクリックし、ウィンドウを閉じます。
- 制御パネル  を開いて、[ データセット ] タブをクリックし、すでにあるデータセット *base* をダブルクリックして削除し、*cfc11jap* だけを残します。
- 「大気中のフロン放出」をクリックし、ワークベンチ変数とします。グラフ  をクリックします。

データセットにもとづき、「大気中のフロン放出」のグラフが表示されます。




- [ Del ] を押すか、もしくは閉じるボタンをクリックして、グラフを閉じます。



このシミュレーションは、1992 年までしか実行されません。なぜなら予測値が、入力したデータセットに含まれていないためです。

### 16.5.1 シミュレーション


データ変数には、データソースが必要です。これらのデータソースは、読み込んだデータセットファイルで、シミュレーションが実行される前に指定する必要があります。デー

タソースを設定するには、シミュレーション制御ダイアログを用いるか、シミュレーションの準備  によって表示されるツールバーを用います。

### 16.5.2 ツールバー


- シミュレーションの準備  を押します。
- シミュレーションの実行名 (例えば *baserun*) を入力します。
- 左の編集ボックス (積分法 [ Euler ] 【オイラー】の右側) に、変換したデータセットの名前 *cfc11jap* を入力します。もしくは、編集ボックスの右にある小さなボタン  を押して、データセットを選びます。

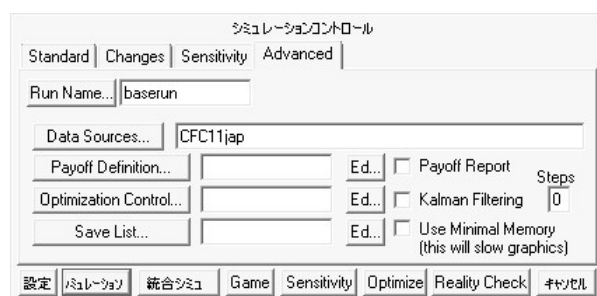


- シミュレーションの実行  をクリックします。

シミュレーションデータで使用するデータセットを指定するもう 1 つの方法は、シミュレーション制御を使うことです。

### 16.5.3 シミュレーション制御 (PLE と PLE Plus は対応しません)

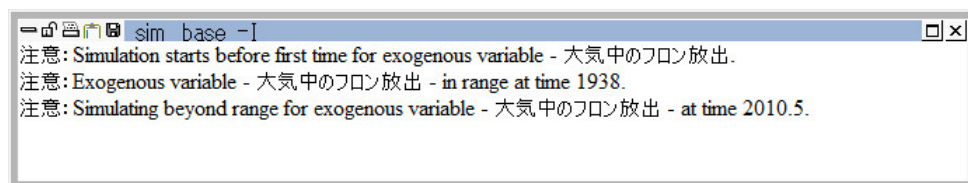
- シミュレーション制御  をクリックし、[ Advanced ] タブをクリックします。



[ Advanced ] タブは、データファイルとモデルの最適化の設定を含んでいます。


- シミュレーションの実行名を入力します (例えば *baserun*)
- [ Data Sources... ] をクリックして、データセット *cfc11jap.vdf* をクリックし、[ 開く ] をクリックします (もしくは、[ Data Source ] の右側の編集ボックスに *cfc11jap.vdf* と入力します)
- [ シミュレーション ] をクリックします。

3 つのエラーが表示されます。



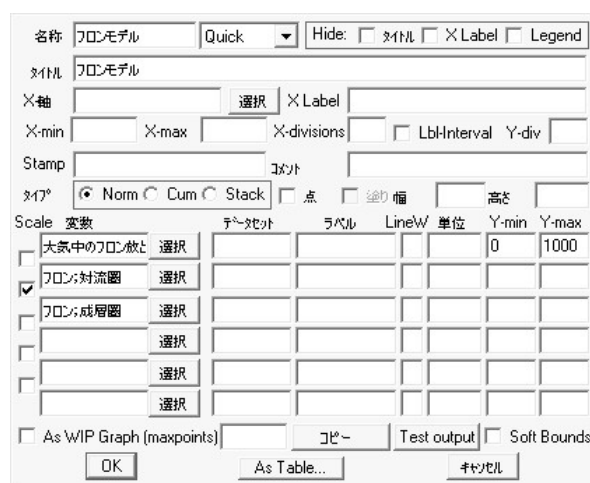
シミュレーションがデータの範囲を超えた場合、最も近い日付の値が使用されます。例えば 1938 年以前では、「大気中のフロン放出」の最初の値が使用され、1994 年以降は、その最後の値が使用されます。

#### 16.5.4 結果

➤ 制御パネル  をクリックし、[ データセット ] タブの中の、*cfc11jap* をダブルクリックします。



モデルは、シミュレーション中に、データセット *cfc11jap* は、変数にデータを提供するために用いられました。シミュレーションにより作成された *baserun* というデータセットは、このデータ変数も含む値を持っています。

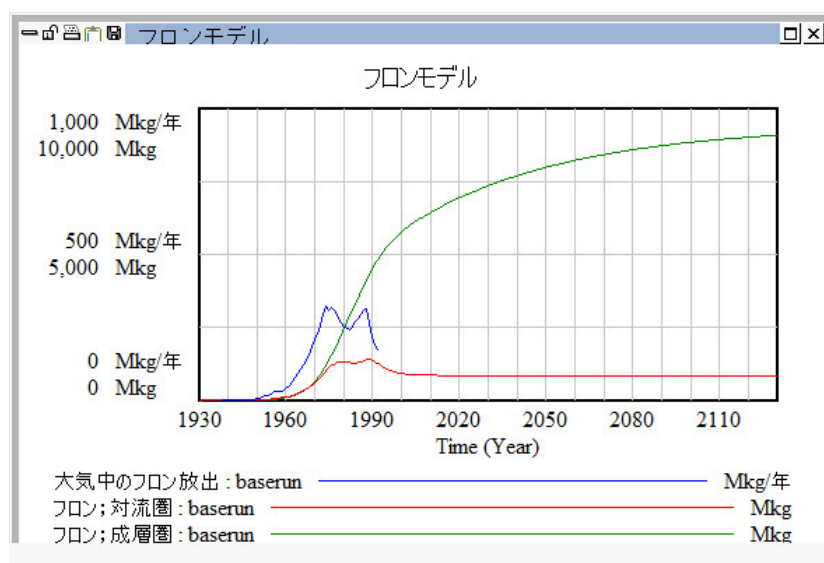
独自のモデルを作成している場合、以下のようにカスタムグラフを定義します。



➤ [ グラフ ] タブをクリックし、[ 表示 ] をクリックしてカスタムグラフを表示します。

もしくは以下のようにします。

➤ グラフ  と直接原因グラフ  を使って、このモデルのフローとストックの振る舞いを見ます。



読み込まれた時系列データの「大気中のフロン放出」は1992年で終わっています。しかし、その最終値の171.1がそれ以降も用いられることを意味しています。このため、「フロン; 成層圏」が上昇を続けている一方で、2000年以降「フロン; 対流圏」は安定します。

フロンの消費と放出は、引き続き何年かにわたって減少し続けるかもしれません。フロンの放出が、171.1 Mkg/年と同じ水準で続くことはないでしょう。フロン放出に対する楽観的な予測を持った、時系列データを開いてみます。

## 16.6 表計算データの読み込み (PLE は対応しません)

- 表計算アプリケーション (Lotus 123 や Microsoft Excel など) があれば、*cfc11jap.wkl* もしくは *cfc11jap.xls* を開いて、データの内容を見ます。

表計算ファイルの一部は以下のようになります。

	B	C	D	E	F	G
1	Model Variables	Historical Data (see Source, below)				
2		1938	1939	1940	1941	1942
3						
4	大気中のフロン放出	0.1	0.1	0.1	0.1	0.1
5		0.1	0.1	0.1	0.1	0.1
6		0	0	0	0	0
7		0	0	0	0	0

- [モデル] メニュー [データセット読込] を選択し、*cfc11jap.xls* もしくは *cfc11jap.wkl* を選んで [開く] をクリックします。

VDF 変換の編集ダイアログが開きます。

最初の 2 行 3 列はシートの  
“C2” セルに、4 行 61 列は  
“BI4” セルに相当します。

時間データや変数データ及び変数名を正確に読み込むために、オプションを設定する必要があります。まず全てのデータを含むセルの範囲を設定します。

- 左側の [ 行 # ] の編集ボックスに「2」を入力します。
- そのとなりの [ 列 # ] の編集ボックスに「3」を入力します。
- 次の [ 行 # ] の編集ボックスに「4」を入力します。
- 右端の [ 列 # ] が「61」になっていることを確認します。

変数名「Time」がないので、時間の値を読み込む行を指定します。

- [ Variable is time axis ] 【時間軸の変数】と同じ行にある [ 行 # ] の編集ボックスに「2」を入力します。

[ Variable is time axis ] のオプションボタンはオフとなり、[ 行 # ] のボタンはオンになります。変数名のデフォルト位置は 1 列目です。モデル変数名「大気中のフロン放出」は 2 列目にあるので、これを指定します。表計算の 3 行目は、データを含んでいないため、除外します。

- [ Var: ] の右にある [ 列 # ] を「2」とします。
- [ List of rows to exclude ] 【除外する行】の編集ボックスをクリックし、「3」を入力します。
- [ Save Format Information... ] 【形成の保存】をクリックし、名前を *cfc11\_jap* と入力し、[ 保存 ] をクリックします。

同じデータファイルを再度変換したい場合、このフォーマット情報を読み込むことができます。VDF 変換の編集ダイアログは以下のようになります。



Table to VDF conversion for: CFC11.XLS

範囲 ☐ 全て 行# 2 列# 3 から 行# 4 列# 61

時間軸名 Time ☒ Across ☐ Down Time values recognized when:

☐ Variable is time axis or ☒ 行# 2 or ☐ Formula 0 + 1 per col

Var: ☒ 列# 1 or ☐ 行# 1 Subs: [ ]

☒ Strip "" Value for empty cells [ ]

List of rows to exclude: e.g. 6,9,33 [ 3 ]

List of columns to exclude: [ ]

Translation Control

[ ] Mov Sel [ ] Ed Sel [ ] Add Ed [ ]

Load Format Information ... [ ]

Save Format Information ... [ ]

OK Translate キャンセル

Contents View

列 1	列 1
001  Atmospheric	001  Atmospheric
002  Model Varial	002  Year
003  Historical D	003  --
004  --	004  atmospheric
005  --	005  Refrigerator
006  --	006  Blowing age
007  --	007  Open-cell fo
008  --	008  --
009  --	009  --
010  --	010  Source: Alt
011  --	011  *** Wz


➤ VDF 変換の編集ダイアログの [ OK ] をクリックします。

[ cfc11jap は存在します。上書きしますか？ ] という表示が現れます。

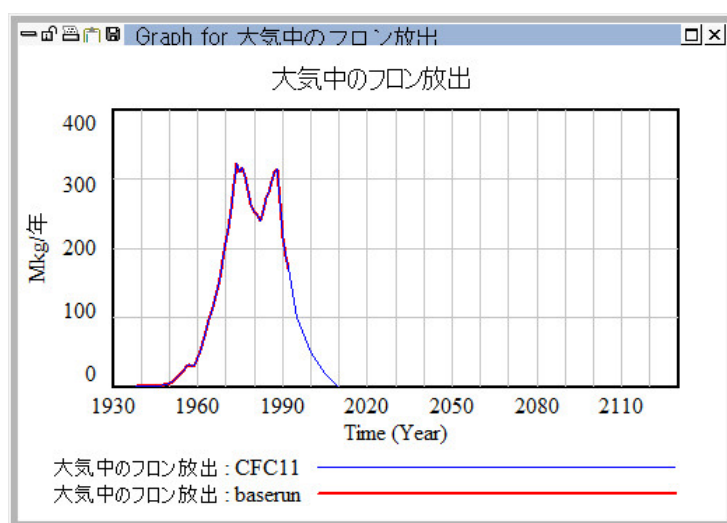
➤ [ はい ] をクリックします。

[ Conversion to vdf completed without error ] というメッセージが表示され、操作した内容を表示する出力画面が開きます。


➤ メッセージボックスの [ OK ] をクリックし、ウィンドウを閉じます。

➤ 「大気中のフロン放出」をクリックし、グラフ  をクリックします。

「大気中のフロン放出」のグラフは、データセット *cfc11jap* では、ゼロまで減少していきます (データセット *baserun* のグラフは、1992 年までは、これと重なっています)。





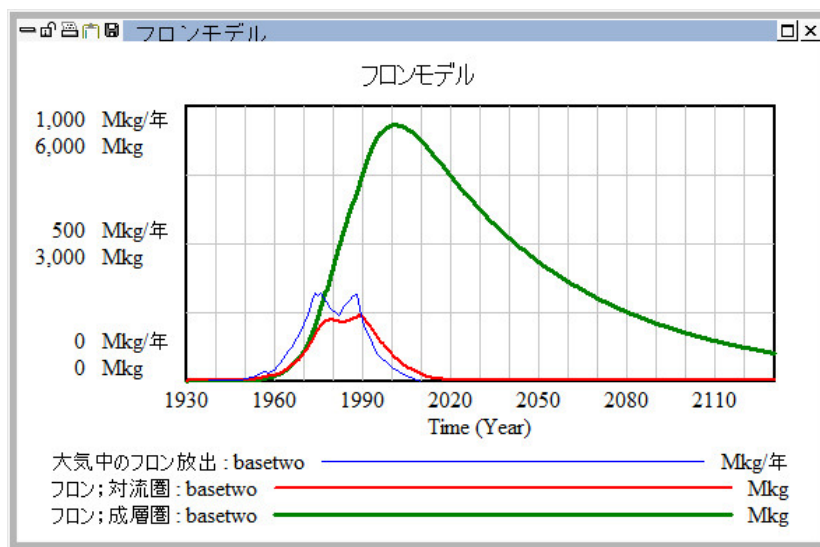
### 16.6.1 シミュレーション

- シミュレーションの準備  をクリックします。実行名を入力します（例えば *basetwo* ）。

「大気中のフロン放出」のデータには、このデータセットへの参照が必要です。

データセット *cfc11jap* は、まだデータ編集ボックスに表示されています。

- シミュレーションの実行  をクリックします。
- 制御パネル  をクリックします。[データセット] タブをクリックし、データセット「*cfc11jap*」と「*baserun*」をダブルクリックし、取り除きます。[グラフ] タブをクリックし、[表示] をクリックします。



フロンの放出が停止することで、「フロン；成層圏」の量が大幅に減少していることがわかります。しかし、それはかなりの遅れをもって成層圏に影響します。「大気中のフロン放出」が、1988年に減少し始めても、「フロン；成層圏」の値は、2002年に頂点に達します。さらに、2060年までは、「フロン；成層圏」の値は1988年の値を上回ります。このモデルは非常に単純で、フロン放出とオゾン層破壊の完全な関係を示すものではありません。しかし、現在の活動がもたらす結果を実感するためには時間が必要であることを示しています。

## 第 17 章

# 下添え字

下添え字は、Vensim Professional と DSS のみ対応します。下添え字のあるモデルは、Vensim Model Reader で利用できます。

### 17.1 下添え字とは

下添え字は、1 つの変数や方程式が、いくつかの異なる概念を表します。例えば、以下のような方程式を考えます。

$$\text{利益} = \text{収入} - \text{経費}$$

もしチェーン店のモデルを作成した場合、それぞれの店の活動に興味が生じます。この時、「店」を下添え字として定義することができます。

店：東京、大阪、名古屋、福岡、札幌、那覇

それぞれの名前は、異なる店（この場合は場所です）を示しています。方程式は次のように書けます。

$$\text{利益}[\text{店}] = \text{収入}[\text{店}] - \text{経費}[\text{店}]$$

この方程式は、それぞれの店について示しており、その店の利益は、その店の収入から、その店の経費を差し引くことにより計算されます。したがって、6 つの方程式ではなく、1 つの方程式だけで記述できます。さらに、もし店を追加したい場合でも、方程式を書きなおす必要はありません。

モデルにある変数は、異なる下添え字を持っていたり、また下添え字を持っていなかったりします。例えば、以下のような方程式があります。

$$\text{企業利益} = \text{SUM}(\text{利益}[\text{店!}]) - \text{企業経費}$$

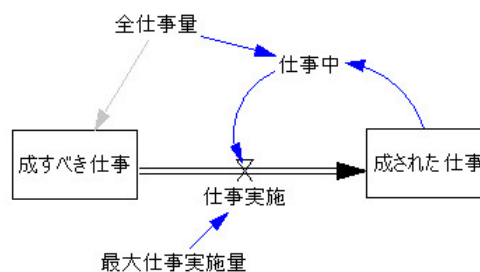
この方程式はすべての店の「利益」を合計し、「企業経費」を差し引きます。ここでの要点は、下添え字と通常の変数を混在させて計算できることであり、また、それが必要となる場合があることがわかります。

## 17.2 簡単なプロジェクトモデル (proj1.mdl)

このプロジェクトモデルは、3 つの仕事から構成されています。各仕事は同じ構造を持ちます。proj1.mdl は 1 つの仕事モデル化しています。このモデルを、proj2.mdl として保存し、3 つの仕事を示す下添え字を持たせます。

英語版です。

- guide\chap17 にある proj1.mdl を開きます。
- もしくは、以下のような図と方程式のモデルを作成します。[ 時間の範囲 ] は、[ 開始時間 ] = 「0」、[ 終了時間 ] = 「100」、[ 時間ステップ ] = 「0.125」とし、[ 時間単位 ] は「月」とします。



### 17.2.1 proj1.mdl 方程式

最大仕事実施量 = 8  
Units : 作業/月

仕事中 = IF THEN ELSE(成された仕事 < 全仕事量, 1, 0)  
Units : Dmnl

全仕事量 = 180  
Units : 作業

仕事実施 = 最大仕事実施量 \* 仕事中  
Units : 作業/月

成された仕事 = INTEG(仕事実施, 0)  
Units : 作業


成すべき仕事 = INTEG (- 仕事実施, 全仕事量)  
Units : 作業

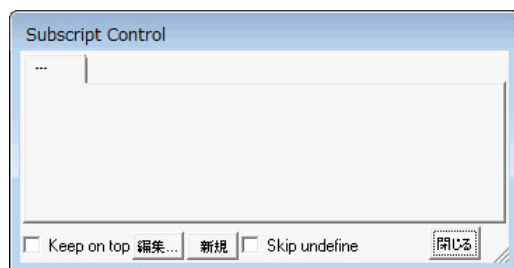
- 「成すべき仕事」の方程式編集ダイアログで、[ 補充的に使用 ] にチェックを入れます。
- モデルの振る舞いを検討するために、シミュレーションを実行します。「成された仕事」が、ゼロから最終の値まで、一直線に増加し、その後一定になることがわかります。
- [ ファイル ] メニュー [ 別名で保存 ] を選択し、proj2 として保存します。

## 17.3 複数の仕事を持つプロジェクト (proj2.mdl)

### 17.3.1 下添え字範囲

まず、複数の仕事を持つことのできる変数を作成します。これは、下添え字範囲と呼ばれ、この場合で適切な名前は「仕事」です。

- ツールバーの下添え字  をクリックし、下添え字コントロールを開きます。



- [ 新規 ] をクリックし、名前を「仕事」と入力し、[ OK ] をクリックします。

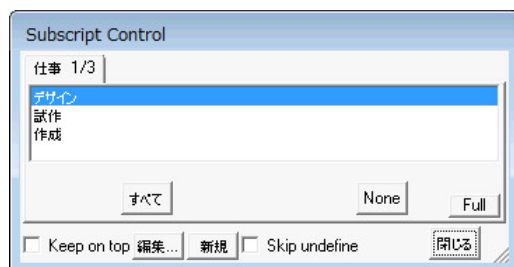
「仕事」の方程式編集ダイアログが開きます。これは下添え字範囲を示し、デザイン、試作、作成の 3 つの仕事を加えます。「仕事」という下添え字の範囲は、これらの具体的な仕事内容と同じではなく、仕事を表すものです。方程式編集ダイアログの、編集ボックスの左に、下添え字の記号 : が表示されることに注意してください。

- 3 つの仕事を以下のように入力し、([ 仕事: ] と入力する必要はありません。すでに表示されています)[ OK ] をクリックします。

仕事 : デザイン, 試作, 作成

3 つの仕事が下添え字コントロールに表示されます。下添え字範囲「仕事」と 3 つの下添え字はスケッチに表示されないことに注意してください。下添え字範囲と下添え字は、モデル定義の一部であり、図における構造の一部ではありません。

### 17.3.2 下添え字コントロール




下添え字範囲の名前「仕事」のタブが付いたリストボックスに、3 つの下添え字の要素が表示されます。始めの要素 [ デザイン ] が強調表示されています。分析ツールは、下添え字範囲の中で選択された（強調表示された）下添え字の要素だけに機能します。下添え字の要素を強調表示させるには、それをクリックします。[ すべて ] をクリックすると、すべての要素が強調表示され、[ None ] をクリックすると、すべての強調表示を解除することができます。[ 編集 ] をクリックすることで、方程式編集ダイアログの中で、いつでも下添え字範囲を編集できます。

[ Full ] ボタンでは、多数の（12 を超える）要素を持つ下添え字範囲を作成するための、より詳細な設定ができます。下添え字コントロールの詳細は、リファレンスマニュアルの 12 章で紹介しています。

### 17.3.3 変数への下添え字の追加


変数に下添え字を追加する場合、まず、どの変数がどの下添え字を利用するかを決める必要があります。このモデルに関しては、これまで定義されたすべての変数に下添え字を追加することが賢明です。

- スケッチ上をクリックするか、またはツールバーのモデルウィンドウ表示/入れ替え  をクリックします。[ 編集 ] メニュー [ 全て選択 ] を選択するか、もしくは **Ctrl** + **A** によりすべての変数を強調表示させます。
- [ 編集 ] メニュー [ 下添え字の設定 ] を選択し、下添え字編集ダイアログを開きます。



- 最初のドロップダウンボックス ([ 下添え字 1 ]) をクリックし、[ 仕事 ] を選択します ([ --none ] と [ 仕事 ] の 2 つの選択肢があります)。
- [ OK ] をクリックします。

モデルの全ての変数が下添え字「仕事」を持つようになります。スケッチ上では、これらの下添え字は表示されないことに注意してください。下添え字は、変数を選択し、ワークベンチ変数とした時、タイトルバーに表示されます。また、方程式でも見ることができます。

- 方程式  を選択し、「仕事実施」をクリックします。方程式は次のようになります。

$$\text{仕事実施 [仕事]} = \text{最大仕事実施量 [仕事]} * \text{仕事} [\text{仕事}]$$

下添え字が、Vensim 方程式の角括弧 [] の中表示されます。もし、変数に 2 つ以上の下添え字がある場合、異なる下添え字はコンマによって区別されます。例えば、在庫 [備蓄, 品目] のようにします。

最も重要な下添え字の規則は、方程式の右項に表示されている下添え字は、左項にも表示されるということです。このモデルについては、これは問題ではありません。

➤ [キャンセル] をクリックし、方程式編集ダイアログを閉じます。

### 17.3.4 定数の扱い

ここまでの修正で、モデルの全ての方程式は同じです。特に、「全仕事量」の方程式は以下のようにになっています。

$$\text{全仕事量 [仕事]} = 180$$

この方程式は、各「仕事」(つまり、デザイン、試作、作成) が 180 という値であることを示しています。下添え字の目的の 1 つは、異なる要素が、異なる値を持つことを可能にすることです。これは、後の節において紹介する多元方程式を利用することができます。特に定数については、コンマで区切って入力するだけで作成できます。

➤ 「全仕事量」の方程式編集ダイアログを開きます。方程式を、以下のように書き換え、[OK] をクリックします。




$$\text{全仕事量 [仕事]} = 180, 250, 500$$

➤ 「最大仕事実施量」についても、以下のように入力します。

$$\text{最大仕事実施量 [仕事]} = 8, 10, 15$$

### 17.3.5 ベクトル関数

全ての仕事における「成された仕事」の合計を示す変数を追加します。

- 変数  を選択します。「成された仕事」の下をクリックし、名前を「成された仕事の合計」と入力します。
- 矢印  を選択し、「成された仕事」をクリックし、次に「成された仕事の合計」をクリックします。
- 方程式  により、「成された仕事の合計」の方程式編集ダイアログを開きます。
- 変数のリストの中の、「成された仕事」をクリックします。

下添え字も一緒に追加されるので、方程式は以下ようになります。

$$\text{成された仕事量の合計} = \text{成された仕事 [仕事]}$$

この方程式は完成ではありません。この時点でモデルをチェックすると、[Subscript Range -task- appears on the right but not left. 【下添え字範囲 [仕事] が右項にはありますが、左項にはありません】というエラーメッセージが表示されます。「成された仕

事」は下添え字「仕事」を必要としますが、「成された仕事の合計」には下添え字が必要ありません。なぜなら、それは全ての仕事の合計値だからです。こうしたことを実現するために、SUM 関数を使用します。

成された仕事の合計 = SUM(成された仕事「仕事!」)

感嘆符(!)は、どの下添え字範囲を合計したいかを示します。この場合、下添え字は 1 つだけですが、1 つの変数につき、8 つまでの下添え字が可能です。したがって、どの下添え字に対して SUM 関数が作用しているのかを示すことが重要です。

- 上記の方程式の、SUM 関数と下添え字を入力します。「単位」は「作業」とします。編集ボックスの左にある「補充的に使用」にチェックを入れます。感嘆符(!)を忘れないようにします。「OK」をクリックします。

### 17.3.6 方程式編集ダイアログによる下添え字の作成

「編集」メニュー「下添え字の設定」を使用する以外に、方程式編集ダイアログを使用して、変数に下添え字を追加することもできます。「下添え字」タブを選択し、下添え字を追加する変数の名の最後にカーソルを置き、下添え字の名前をクリックします。Vensim は、自動的に角括弧[]もしくはコンマを追加します。「下添え字」タブは、モデルが下添え字を持つ場合のみ、表示されます。下添え字の名前を、直接入力することもできます。このモデルの方程式は、以下のようになります。

仕事 : デザイン, 試作, 作成

最大仕事実施量「仕事」 = 8,10,15  
Units : 作業/月

仕事「仕事」 = IF THEN ELSE( 成された仕事「仕事」 <  
全仕事量「仕事」, 1, 0)  
Units : Dmnl

全仕事量「仕事」 = 180,250,500  
Units : 作業

仕事実施「仕事」 = 最大仕事実施量「仕事」 \* 仕事「仕事」  
Units : 作業/月

成された仕事「仕事」 = INTEG(仕事実施「仕事」, 0)  
Units : 作業

成すべき仕事「仕事」 = INTEG (- 仕事実施「仕事」, 全仕事量「仕事」)  
Units : 作業

すべての成された仕事 = SUM(成された仕事「仕事!」)  
Units : 作業

### 17.3.7 下添え字の要約

最後の 1 つを除いて、方程式の左項には、下添え字が表示されます。定数では、下添え字「仕事」が左項に表示され、右項に 3 つの値(それぞれの仕事内容ごとに 1 つずつ)が表示されます。補助変数では、下添え字「仕事」が、左右どちらにも表示されます。その








値は、仕事ごとに計算されます。

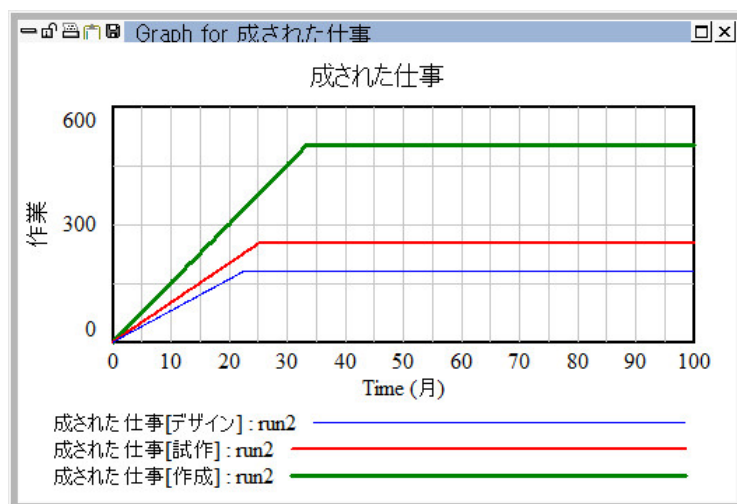
例外が最後の方程式です。「仕事」が右項にのみ表示されます。変数「すべての成された仕事」は、3つの仕事からなる「成された仕事」の合計値で、値は1つです。下添え字の関数 SUM は、感嘆符 (!) が付いた下添え字を持つ変数の値を合計します。方程式に、2つ以上の下添え字の範囲を用いることがあり、それを区別するために、感嘆符 (!) を付けた下添え字が使用されます。

➤ [モデル]メニュー [モデルチェック] を選択します (もしくは **Ctrl+T**)。


[モデル O.K です] というメッセージが表示されます。そうでなければ、エラーメッセージが現れます。方程式編集ダイアログが開き、間違った方程式を示します。方程式が上記の方程式と一致しているかチェックします。その方程式が一致していれば、他の方程式もチェックします。修正し、[OK] をクリックし、[モデル]メニュー [モデルチェック] を再度選択します。

### 17.3.8 シミュレーションと分析

- シミュレーション名の編集ボックス  をダブルクリックし、データセット名を「run2」とします。シミュレーションの実行  をクリックします。モデルをシミュレーションしたあと、制御パネル  の [データセット] タブで、「run2」以外のデータセットを除外します。
- 「成された仕事」をクリックし、グラフ  をクリックします。下添え字「デザイン」の「成された仕事」の振る舞いを表すグラフが表示されます。
- ツールバーの下添え字  をクリックします。下添え字コントロールが開きます。[すべて] をクリックし、再度グラフ  をクリックします。



すべての下添え字が選択されたため、それぞれの仕事ごとの「成された仕事」がグラフに示されます。どの分析ツールも、下添え字コントロールで選択された下添え字を反映した出力を生成します。

➤ 「すべての成された仕事」をクリックし、直接原因グラフ  をクリックします。


3 つの仕事ごとの「すべての成された仕事」の振る舞いと、その原因となる「成された仕事」のグラフが表示されます。この場合、選択された下添え字の数は問題ではなく、すべての下添え字が、「すべての成された仕事」の原因となるため、表示されます。

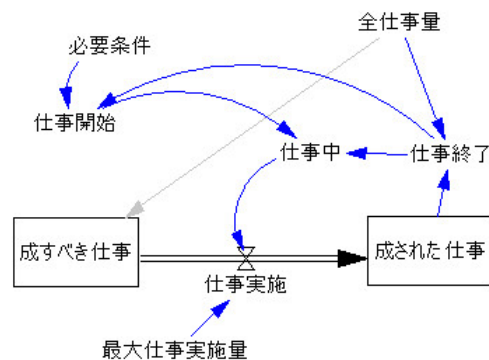
## 17.4 下添え字モデルの応用 (proj3.mdl)

このプロジェクトモデルは、3 つの部門からなる仕事を持つプロジェクトを完成させます。特にある仕事が、次の仕事が始まる前に完了しておく必要があるという条件を与えます。つまり、それぞれの仕事の完成は、後に続く仕事を開始するための必要条件となります。それぞれの仕事の構造は、すべて同じです。


➤ *guide\chap17\complete* から、*proj3.mdl* を開きます。[ファイル]メニュー [別名で保存] を選択し、モデルを *guide\chap17* または、作業用のディレクトリに保存します。これは完成したモデルです。

もしくは、以下のようにします。

➤ 新規モデル  をクリックします。[時間ステップ] を [0.125] とし、[OK] をクリックします。以下のモデルを作成し、*proj3.mdl* として保存します。




これはモデルを見やすくするための作業であり、必須のものではありません。

➤ 下層に隠す  を選択し、「全仕事量」から「成すべき仕事」への矢印の矢頭をクリックし、矢印を隠します（なぜならストックを初期値として利用されているだけだからです）。

### 17.4.1 下添え字範囲

まず仕事を表す下添え字の範囲を作成します。

➤ 下添え字  をクリックします。  
 ➤ [新規] をクリックし、名前を [仕事] と入力し、[OK] をクリックします。「仕事」の方程式編集ダイアログが開きます。

- 3つの仕事を以下のように入力し、[OK]をクリックします。

仕事 : デザイン, 試作, 作成

編集ボックスに入力するのは「デザイン, 試作, 作成」です。

- 方程式  $f(x)$  を選択し、「全仕事量」の方程式編集ダイアログを開きます。
- 「180,250,500」と入力します。
- 編集ボックスにある「全仕事量」の右側をクリックします。[下添え字] タブをクリックし、「仕事」をクリックします(リストの中は1つだけです)。[ ]と一緒に下添え字が追加されます。方程式は、以下のようになります。

全仕事量 [仕事] = 180,250,500  
Units : 作業

- 他の方程式の入力を続けます。

[変数] タブと [下添え字] タブを交互にクリックする必要があります。一度変数に下添え字を追加すると、[変数] タブの中でその変数をクリックした時、下添え字も一緒に方程式に入力されます。つまり、それぞれの変数は、方程式を作成する際に、下添え字を一度だけ追加する必要があるということです。方程式を以下のように入力します。

成すべき仕事 [仕事] = INTEG ( - 仕事実施 [仕事], 全仕事量 [仕事] )  
Units : 作業

- 「成すべき仕事」の [ 補充的に使用 ] にチェックを入れます。

成された仕事 [仕事] = INTEG(仕事実施 [仕事], 0)  
Units : 作業

仕事実施 [仕事] = IF THEN ELSE(作事中 [仕事], 最大仕事実施量 [仕事], 0)  
Units : 作業/月

「仕事実施」の方程式は、論理関数である IF THEN ELSE が使われています。「作事中」である限り、「最大仕事実施量」の作業が行われます。

最大仕事実施量 [仕事] = 8,10,15  
Units : 作業/月

残った4つの変数は、仕事の関係を表します。仕事が、必要条件を持たない場合、すぐに作業が始まります。仕事が必要条件を持つ場合、その条件が作業開始までに終了している必要があります。

### 17.4.2 下添え字の写像

仕事は、(1) 独立した仕事、もしくは(2) 他の仕事が終了するという必要条件を持つ仕事のいずれかです。両方のコンセプトを1つの方程式に表示するために、新たに下添え字範囲「前仕事」を定義します。これは、これまでの下添え字範囲「仕事」と全く同じです。この過程を写像と言います。このプロジェクトモデルでは、それぞれの仕事は、基本的に他の仕事が終了することを必要条件としています。したがって、最初に作成した下添え字範囲を写像記号 < - > を使用して、新たな下添え字範囲を定義します。

写像はマッピングともいいます。

- 下添え字  $x_i$  をクリックします。[新規] をクリックし、名前を「前仕事」と入力し、[OK] をクリックします。

- [タイプ] の下のドロップダウンボックス ( ) で、[ Equivalence ]【同等】を選択します。

方程式編集ダイアログの、編集ボックスの左に、同等を表す記号  $\sim$  が表示されます。

- 方程式編集ボックスに、[ 仕事 ] と入力します。括弧は表示されません。
- [ OK ] をクリックし、方程式編集ダイアログを閉じます。

副範囲は、サブレンジともいいます。

下添え字制御ダイアログに、新たなタブが追加されていないことがわかります。Equivalence【同等】は、Subrange【副範囲】とみなされます。つまり、「前仕事」は「仕事」の副範囲であり、個別のタブを持ちません。もし、「前仕事」を編集したい場合は、方程式編集ダイアログの [ 選択 ] ボタンを利用します。

### 17.4.3 下添え字を持つ変数の多元方程式

仕事は、必要条件を持ちます。しかし、それ自身の必要条件ではありません。3 つの仕事（デザイン、試作、作成）について、どの仕事か、他の必要条件であるかを示す方程式（3 つの仕事がそれぞれの仕事に対して、合計 9 つ）の設定をする必要があります。例えば、「試作」では、次のような方程式を設定します。

$$\begin{aligned} \text{必要条件 [試作, デザイン]} &= 1 \\ \text{必要条件 [試作, 試作]} &= 0 \\ \text{必要条件 [試作, 作成]} &= 0 \end{aligned}$$

「試作」に関する 3 つの方程式は、デザインは必要条件（値 = 1）です。しかし、試作と作成は必要条件ではない（値 = 0）ことを示しています。「試作」のこれら 3 つの方程式は、より簡潔な形にすることができます。

$$\text{必要条件 [試作, 前仕事]} = 1, 0, 0$$

必要条件の仕事の順序が、下添え字として作成された仕事と同じことに注意してください。そのため、値 1 はデザインを、値 0 は試作を意味しています。しかし、全体で 3 つの方程式が必要です（必要条件を表すために、2 次元配列を持つ、1 つの方程式で表すこともできます。これはリファレンスマニュアルを参照してください）。

- 「必要条件」をクリックします。変数名のすぐ後ろをクリックし、カーソルを置きます。
- [ 下添え字 ] タブのドロップダウンボックス ( ) ([ Range ] となっています) をクリックします。ドロップダウンリストから、下添え字範囲 [ 仕事-elements ] を選択します。下のリストボックスに 3 つの仕事が表示されます。
- リストから、下添え字「デザイン」をクリックします。これで、方程式のカーソルを置いたところに下添え字が入力されます。方程式に表示された [ デザイン ] の文字と ] の間をクリックし、カーソルを置きます。
- ドロップダウンボックス ( ) ([ 仕事-elements ] となっています) をクリックします。ドロップダウンリストから [ Range ] をクリックします。リストボックスは、[ Range ] と 2 つの下添え字範囲が表示されます。

- 「前仕事」をクリックします。
- 方程式編集ボックス（等号の後）をクリックし、カーソルを移動させ、数字とコンマを以下のように入力します。

必要条件 [デザイン, 前仕事] = 0,0,0  
Units : Dmnl

これは、どの仕事が他の仕事の必要条件かを示す 3 つの方程式の一番目です。次の 2 つの方程式を作成します。

- 方程式編集ダイアログの右上にある [ 式追加 ] ボタンをクリックします。

「必要条件」に 2 つ目の方程式が追加されます。方程式編集ダイアログを閉じ、再度開いていたら、自動的に下添え字 [ 仕事, 仕事 ] が追加されています。「デザイン」は仕事で、「前仕事」も仕事への写像なので、先程の方程式と同じ形です。もし方程式編集ダイアログで続けて作業しているなら、「必要条件」は下添え字を持たずに表示されます。

- 「必要条件」に下添え字があれば、削除します。
- 新しい下添え字を入力するか、もしくは下添え字のリストから選択して、以下の方程式を完成させます。

必要条件 [試作, 前仕事] = 1,0,0

[ 単位 ] はすでに入っていることに注意してください。変数に定義された単位は 1 つです。

- 右上のドロップダウンボックス( ) ( [ 2 ] となっています ) をクリックし、[ \*New\* ] をクリックします。新しい下添え字を入力するか、もしくは下添え字のリストから選択して、以下の方程式を完成させます。

必要条件 [作成, 前仕事] = 1,1,0  
Units : Dmnl

[ \*New\* ] を選択したドロップダウンボックス( ) をクリックすることで、3 つの方程式を切り替えることができます。方程式を追加することで、このリストに方程式の番号が追加され、タイトルバーにも表示されます。[ Del ] をクリックすると、表示されている方程式が削除されます。



- [ OK ] をクリックし、方程式編集ダイアログを閉じます。

注意 2 次元配列を用いた、簡潔な方程式もあります。以下の方程式は、「必要条件」に入力した 3 つの方程式と同じ内容です。

必要条件 [仕事, 前仕事] = 0,0,0;1,0,0;1,1,0;



この方程式は VMIN 関数を使用しています。これは、感嘆符 (!) を持つ下添え字で作動します。3 つの仕事と 3 つの必要条件があるため、この方程式は 9 つの値を持ちます。

例えば、「デザイン」を見てみます。「デザイン」は、必要条件を持たず、すべての必要条件の値はゼロになります。

必要条件 [デザイン, 前仕事] = 0,0,0

IF THEN ELSE 関数は、「必要条件 [デザイン]」の 3 つの値を評価し、ゼロを返します。したがって、論理テストは NO となり、1 の値を返す ELSE 文を実行します。これは、シミュレーションが実行されるとすぐに、必要条件を持たない「デザイン」の仕事が開始することを意味します。

次に、「作成」を見てみます。「作成」は、以下のように、必要条件として「デザイン」と「試作」の両方を持ちます。

必要条件 [作成, 前仕事] = 1,1,0

IF THEN ELSE 関数は、「必要条件 [作成, 前仕事]」の値を評価し、3 つの必要条件のうち 2 つで 1 を返したものは THEN 文を実行し、「仕事終了 [前仕事]」を評価します。ここで「前仕事」が「試作」であったとします。「試作」の仕事が終了した場合、「仕事終了」は 1 を返し、そうでなければ 0 を返します。どの必要条件となる仕事も 0 ならば、VMIN 関数も 0 を返し、仕事の開始をさせません。すべての必要条件となる仕事が 1 を返すと、VMIN は 1 を返し、最後に仕事が始まります。


➤ [モデル]メニュー [モデルチェック](または $\text{Ctrl}+\text{T}$ )を選択します)。

➤ [モデル]メニュー [単位チェック](または $\text{Ctrl}+\text{U}$ )を選択します)。


[OK]が表示されます。もしそうでなければ、方程式と構造を、上記のものと比較してください。


#### 17.4.6 シミュレーションと分析


➤ シミュレーション名の編集ボックスをダブルクリックし、データセット名を、「run3」とします。

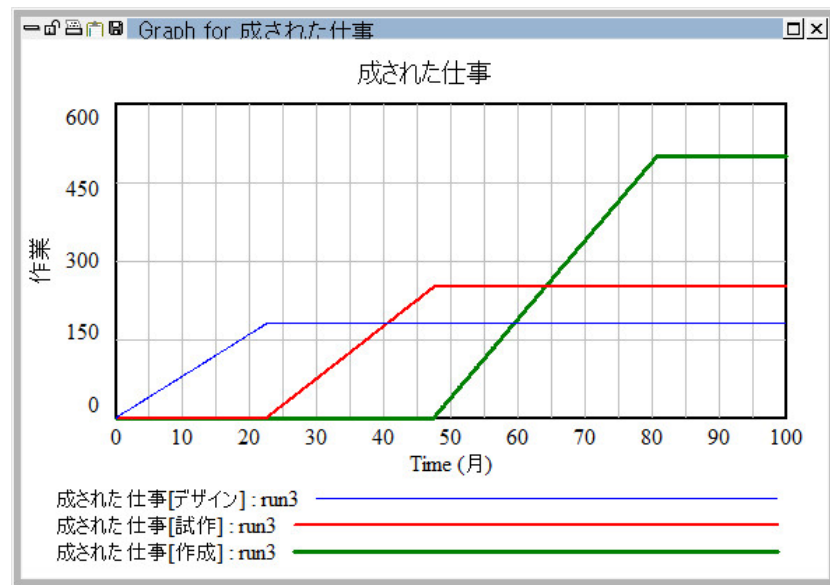
➤ シミュレーションの実行  をクリックします。

モデルがシミュレーションされます。

➤ 「成された仕事」をクリックし、グラフ  をクリックします。下添え字「デザイン」だけの「成された仕事」の振る舞いを示すグラフが表示されます。

➤ 下添え字  をクリックします。下添え字コントロールが開きます。[All]をクリックします。

➤ グラフ  をクリックします。



それぞれの仕事が表示され、「デザイン」を除いて、その必要条件が満たされた後、作業が開始されることがわかります。



## 第 18 章

# 最適化

最適化は Vensim Professional と DSS のみ対応します。

### 18.1 最適化の概要

最適化とは、最良の結果を得ることです。最適化には、(1) パラメータを補正する“キャリブレーション”と(2) 最良の代替案を選択する“方針の最適化”があります。

最適化には、何が良くて、何が悪いのかという定義が必要です。これを、ペイオフ（利得）と呼びます。ペイオフは、シミュレーション結果において定量的に示されます。ペイオフは、モデル全体を 1 つの値にまとめたものもあります。ペイオフ定義ファイルは、拡張子.vpd を持ちます。ペイオフ関数の詳細な説明は、リファレンスマニュアルにあります。

ペイオフを定義したあと、ペイオフを最大にするために、どの定数を変化させるか選択します。その設定は、拡張子.voc を持つ最適化コントロールファイルに保存されます。このファイルは、最適化の方法や、シミュレーション終了の条件も含みます。

最適化の結果は、*runname.out* と呼ばれるファイルに書き込まれます。選択されるオプションによって、他のファイルも作成されるかもしれません。ファイル *runname.out* は、ペイオフを最適化する定数の値を含みます。

注意 最適化は、停止ボタンをクリックするか、もしくは **Esc** を押すことで中断します。

### 18.2 モデルの補正 (electric.mdl)

ここでは、アメリカ商務省のデータを用いて、アメリカにおける世帯の電化普及モデルを補正します。「世帯数」は、時間と共に指数的に増加し、これは、「非電化世帯」と「電化世帯」の両方を増加させます。「非電化世帯」は、「電化世帯」との接触によって、電化に関心を持ち、「電化世帯」へ転換します。「世帯増加」は、「電化割合」によって追加されます。「非電化世帯」がゼロへと減少し、「電化割合」が 1 になると、建設されるすべての新しい世帯が「電化世帯」になります。

モデルの有効性は、実際の時系列データと比較して検証します。モデルが完成し、シミュレーションできる場合、モデルの補正は可能です。補正は、現実のデータにモデルの振る舞いが適合するためのモデル定数の値を見いだします。

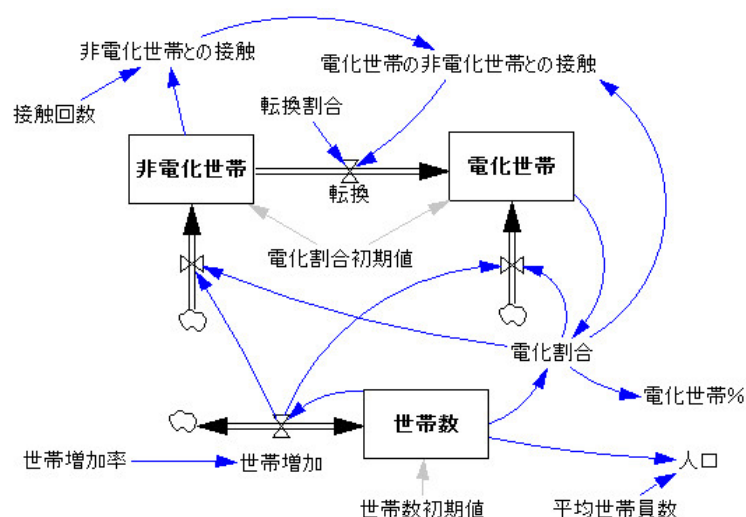
現実のデータとシミュレーション出力が適合するように、モデル定数を手動で変更することは可能です。しかし、最適化させる定数が多く、多くの変数への適合を検討するには、非常に時間を費やします。最適化を使うことで、自動的に選択した定数を変化させて、シミュレーション出力とデータを近づけます。

例えば、「世帯増加率」の値がわからないとします。そこで、このモデル定数を補正するように設定します。最適化の実行により、モデルと実データが近づくようなパラメータを自動的に探します。

➤ `guide\chap18\calibrat` から、`electric.mdl` を開きます。

もしくは、以下のようにします。

➤ 以下のような図と方程式のモデルを作成します。[ 時間の範囲 ] は、[ 開始時間 ] = 「1900」、[ 終了時間 ] = 「1979」、[ 時間ステップ ] = 「0.125」とし、[ 時間単位 ] は「年」です。



### 18.2.1 Electric.mdl 方程式

平均世帯員数 = 4  
Units : 人/世帯

接触回数 = 20  
Units : 接触/世帯/年

転換割合 = 0.01  
Units : 世帯/接触

転換 = 電化世帯の非電化世帯との接触 \* 転換割合  
Units : 世帯/年

電化世帯 = INTEG(転換 + 電化割合 \* 世帯増加, 電化割合初期値 \* 世帯数)  
Units : 世帯

電化世帯の非電化世帯との接触 = 非電化世帯との接触 \* 電化割合

```

Units : 接触/年

電化割合 = 電化世帯/世帯数
Units : Dmnl

世帯増加率 = 0.02
Units : 1/年

電化割合初期値 = 0.05
Units : Dmnl

世帯初期値 = 2e+007
Units : 世帯

世帯増加 = 世帯数 * 世帯増加率
Units : 世帯/年

非電化世帯との接触 = 非電化世帯 * 接触回数
Units : 接触/年

非電化世帯 = INTEG(- 転換 + (1 - 電化割合) * 世帯増加,
(1 - 電化割合初期値) * 世帯数)
Units : 世帯



電化世帯% = 電化割合 * 100
Units : Dmnl

世帯数= INTEG (世帯増加, 世帯数初期値)
Units : 世帯

人口 = 世帯数 * 平均世帯員数
Units : 人

```

方程式の初期値を入力するためには、[ 変数をすべて選択 ] ボタンを使用します。

- モデルを保存します。
- シミュレーション名の編集ボックス  をダブルクリックし、「Current」と入力します。
- シミュレーションの実行  をクリックします。

### 18.2.2 データの比較


データセット *Current* は、初期値の結果を示します。guide\chap18 に、人口と、電化世帯割合を測定したデータがあります。このファイルは、*ELECTRICjap.dat* です。2つの変数「人口」と「電化世帯%」を持ちます。これらのデータをモデルの変数と比較するため、データファイル(.dat)を Vensim データセット(.vdf)に変換します。

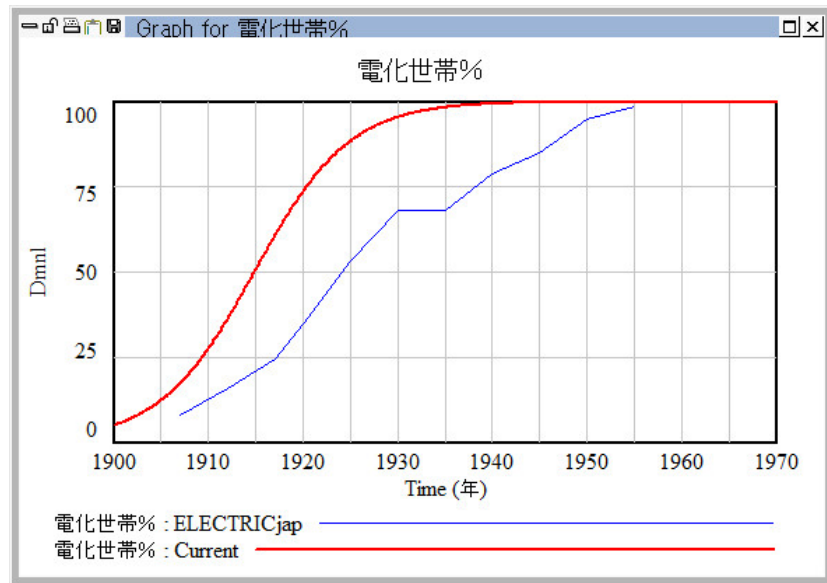
- [モデル]メニュー [データセット読込]を選択し、*ELECTRICjap.dat* をクリックして開きます。

“Dat2vdf completed without error”【エラーなく変換されました】というメッセージが現れ、出力ウィンドウにデータセットにあった変数の数を示します。データセット *ELECTRICjap* は、最初のデータセットとして読み込まれます。したがって、分析ツールはこのデータセットを対象にします。

- メッセージボックスの [OK] をクリックし、出力ウィンドウを閉じます。



元データは、*ELECTRIC.dat* です。エディタでその変数名を日本語化したものを *ELECTRICjap.dat* として保存しました。

- 「電化世帯%」をクリックします。
- グラフ  をクリックします。「電化世帯%」の *Current* と *ELECTRICjap* のデータセットが表示されます。





シミュレーションデータ (Current) は、早く増加し、すぐに頭打ちになります。

### 18.2.3 ペイオフの設定

Vensim には、最適化を設定する 2 つの方法があります。ツールバーの最適化の開始  と、シミュレーションの制御  です。

補正の最適化では、どの変数に適合させるのかを、ペイオフが定義します。ペイオフファイルは、拡張子 *.vpd* (Vensim payoff definition) を持ち、デフォルトでモデルと同じ名前を持ちます (このモデルでは *electric.vpd*)。

- シミュレーション名の編集ボックス  をダブルクリックし、「run2」と入力します。
- ツールバーの最適化の開始  をクリックします。ペイオフを定義する最適化設定のダイアログが開きます。
- [Calibration]【補正】のラジオボタンが選択されていることを確認します。
- [変数]の編集ボックスに、「電化世帯%」と入力します (もしくは、[選択] ボタンをクリックし、リストから選択してください)。
- [ウェイト]の編集ボックスに、「0.25」と入力します。[編集して追加] をクリックします。
- 変数の編集ボックスに「人口」と入力します (もしくは [選択] ボタンをクリックし、リストから選択してください)。
- [ウェイト]の編集ボックスに、「 $3e-7$ 」 (もしくは 0.0000003) と入力します。

$3 \times 10^{-7}$  のことです。

ウェイトは、異なる変数の値にバランスを与えます。最適化のプロセスは数字だけを考慮するので、より大きな値の変数は、小さな値のものより重要とみなされます。ウェイト

に定められた数は、データと測定の単位に基づいています。「電化世帯%」はおよそ 50 の値を持ち、ウェイトを 0.25 に設定しています。およそ 2 億の「人口」は、さらに小さいウェイトを持ちます。これらのウェイトは、これらの変数における予測誤差の標準偏差とほぼ等しくなるように選択されました。こうしたことは、リファレンスマニュアルの 10 章で検討しています。

- [ 追加して編集 ] をクリックします。ペイオフ設定のダイアログは以下のようになります。



- [ 次へ ] をクリックし、最適化コントロールの画面に移ります。

#### 18.2.4 最適化パラメーターの設定

ペイオフが定義されたら、最適化コントロールの設定が必要です。最適化ファイルは、拡張子.voc ( Vensim optimization control ) を持ち、デフォルトでモデルと同じ名前を持ちます ( このモデルでは *electric.voc* )。この例では、パラメーターを求める変数を設定し、その変数の、最大値および最小値の範囲を入力します。

- [ 定数選択 ] をクリックし、[ 転換割合 ] を選択します。最小値 ( <= の左側のボックス ) に「0」と入力し、最大値 ( <= の右側のボックス ) に「0.5」と入力します。[ 編集して追加 ] をクリックします。
- [ 定数選択 ] をクリックし、[ 世帯増加率 ] を選択します。最小値 ( <= の左側のボックス ) に「0」と入力し、最大値 ( <= の右側のボックス ) に「0.1」と入力します。[ 編集して追加 ] をクリックします。
- [ 定数選択 ] をクリックし、[ 電化割合初期値 ] を選択します。最小値 ( <= の左側のボックス ) に「0」と入力し、最大値 ( <= の右側のボックス ) に「1」と入力します。[ 編集して追加 ] をクリックします。
- [ 定数選択 ] をクリックし、[ 世帯数初期値 ] を選択します。最小値 ( <= の左側のボックス ) に「0」と入力し、[ 編集して追加 ] をクリックします。

最適化コントロールの設定は以下のようになります。

Optimization Setup

Optimization Control. Edit the filename to save changes to a different control file

ファイル名: electric.voc [新規ファイルを選択] [設定の削除]

Output Level: On [Trace] [Off] Sensitivity: Off [Off] = [ ]

Multiple Start: Off [Off] Random type: Linear [Seed] [ ]

#Restart: 0 Optimizer: Powell [Max Iterations: 1000] [Max Sims: ]

Pass Limit: 2 Fractional Tolerance: 0.0003 Tolerance Multiplier: 21

Absolute Tolerance: 1 Scale Absolute: 1 Vector Points: 25

Currently active parameters (drag to reorder)

0<=転換割合<=0.5  
0<=世帯増加率<=0.1  
0<=電化割合初期値<=1

[選択して削除]  
[選択して修正]  
[編集して追加]

0 <= 世帯数初期値 = [ ] <= [ ]  
定数のモデル値 2e+007 [定数選択...] [ ] = [ ]

[戻る] [次へ>] [完了] [キャンセル]

- > [次へ] をクリックします。
- > [Data Sources...] をクリックし、データセット *ELECTRICjap.vdf* を選択します。編集ボックスに直接入力することもできます。
- > [完了] をクリックします。

Vensim は、データとモデルを適合させるために、何度もシミュレーションを行います。最適化の結果は、以下のように出力ウィンドウに表示されます。

```

sim_run2 -I -o electric.voc
Initial point of search
  転換割合 = 0.01
  世帯増加率 = 0.02
  電化割合初期値 = 0.05
  世帯数初期値 = 2e+007
Simulations = 1
Pass = 0
Payoff = -1766.51


-----
Maximum payoff found at:
  転換割合 = 0.0055111
  世帯増加率 = 0.0133323
  電化割合初期値 = 0.055457
  *世帯数初期値 = 1.98182e+007
Simulations = 202
Pass = 3
Payoff = -16.8901

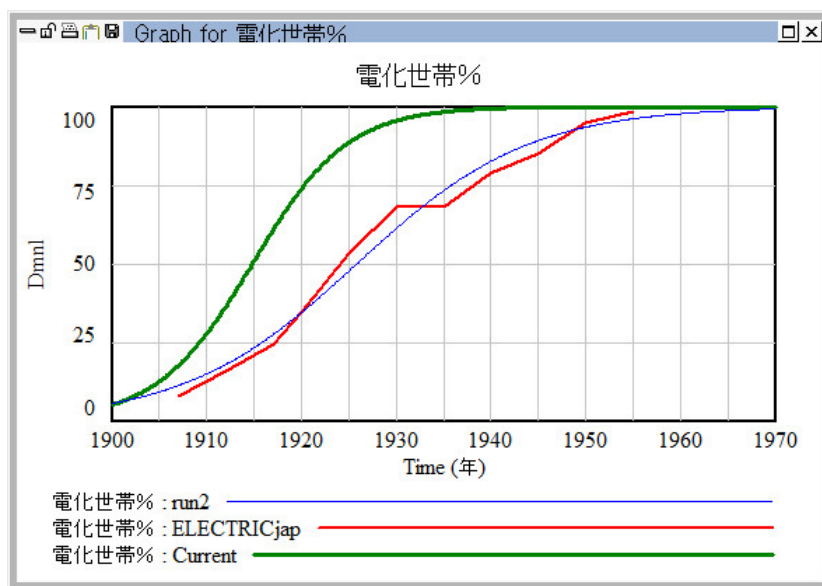
-----
The final payoff is -16.8901

```

ペイオフ最大値は、実データに最も近いモデルを意味します。その時のペイオフの値や定数の値が示されます。またこのデータは、現在のシミュレーションデータセット(*run2*)に保存されます。

また、最適化についての追加の情報は、シミュレーションと同じ名前を持つ *.out* ファイルに保存されます（ここでは *run2.out* ）。


- 「電化世帯%」をクリックし、ワークベンチ変数とします。
- グラフ  をクリックします。「電化世帯%」の *Current,run2,ELECTRICjap* の3つのデータセットが表示されます。




補正された *run2* は、最初のシミュレーション *Current* よりも、*ELECTRICjap* に近づいています。

### 18.2.5 信頼区間の計算

最適化で求められた定数の値には、不確実性があります。最適値の感度分析によってこの不確実性を検討できます。デフォルトのペイオフの感度が 95 % の信頼区間を持つように、ウェイトを選択します。

- ツールバーのシミュレーションの制御  をクリックします。
- 新しいシミュレーション名を「run3」と入力します。
- [ Advanced ] タブをクリックします。
- [ Optimization Control ] の右側にある [ Ed... ] をクリックします。
- [ Sensitivity ] は [ Off ] に設定されています。右側のドロップダウン ( ) をクリックし、[ ペイオフ Value ] を選択します。= の右側に「4」と入力します（ペイオフ = 4）。[ OK ] をクリックします。
- [ Changes ] タブをクリックします。[ Load Changes From ] に「run2.out」と入力します。

最適化の出力は、定数の入力ファイルとして読めるように作成されています。この操作によって、前回の最適化で見いだされた値を利用できます。

- 最適化の開始  をクリックします。
- 最適化が終了したら、[ ファイル ] メニュー [ ファイル編集 ] を選択し、*run3\_sensitiv.tab* を開きます。

このファイルには、パラメータの推定値の 95 % の信頼区間を以下のように示します。

```
:COM The base payoff is -16.8901
:COM A * Means a bound was reached, i.e.payoff not at criterion.
:SENSITIVITY = payoff_VALUE = 4
0.00525701 <= 転換割合 = 0.0055111 <= 0.0057906
0.0128694 <= 世帯増加率 = 0.0133323 <= 0.0137736
0.048541 <= 電化割合初期値 = 0.055457 <= 0.0634687
1.94239e+007 <= 世帯数初期値 = 1.98182e+007 <= 2.02124e+007
```

- [ ファイル ] メニュー [ 閉じる ] を選択し、*run3\_sensitiv.tab* を閉じます。

## 18.2.6 モデルの修正

- 方程式 *f(x)* を選択します。最適化によって与えられた値を反映させるために、モデルの定数を修正します。通常 2～3 ケタの有効数字を用います。

最適値は、以下に示す値か、それに近いものです。

```
転換割合 = 0.0055
世帯増加率 = 0.013
電化割合初期値 = 0.055
世帯数初期値 = 2.0e+007
```

モデルがデータに適合することが、モデルの妥当性を証明するものではないことに注意してください。モデルがデータに適合する場合、データからモデルを不適とすることはできません。モデルの欠陥を見つける他の方法を探さる必要があります。そのためには、本書の 9 章で紹介したリアリティチェックは、大変強力なツールです。

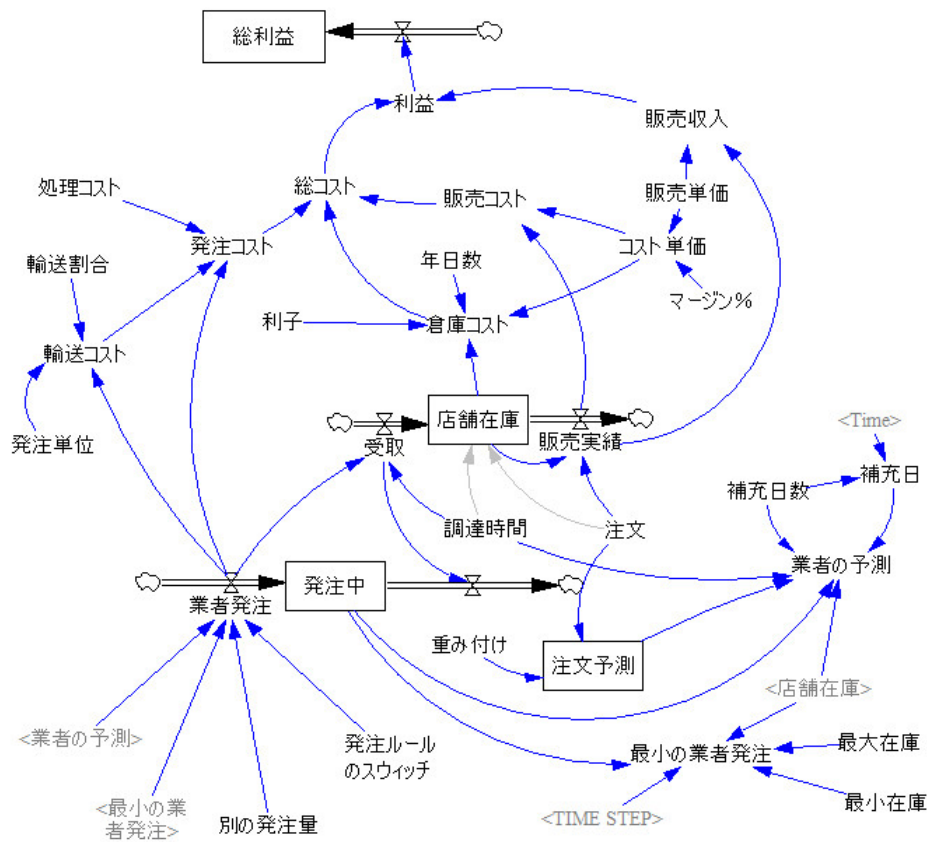
## 18.3 方策の最適化 (stock.mdl)

ここでは、在庫とサプライチェーンを最大利益が得られるように最適化します。モデル *stock.mdl* は、販売によって減少する在庫を持ちます。在庫が少なすぎると、販売の損失（および利益の損失）を招きますが、在庫を多く抱えることは、保管のための経費がかかり、利益の損失を招きます。最大利益を得るためには、在庫の量を調整する必要があります。在庫が少ない場合は、在庫を補充するための生産が必要です。

### 18.3.1 在庫モデル

- *guide\chap18\policy* にある *stock.mdl* を開きます。





FINAL TIME = 60

Units : 日

INITIAL TIME = 0

Units : 日

item : I1111,I2222,I3333

SAVEPER = TIME STEP

Units : 日

TIME STEP = 1

Units : 日

コスト単価 = (1 - マージン% / 100) \* 販売単価

Units : 円/個

マージン% = 10

Units : Dmnl

倉庫コスト = 店舗在庫 \* コスト単価 \* 利子 / 年日数

Units : 円/日

処理コスト = 2

Units : 円/日

別の発注量 = 0

Units : 個/日

利子 = 0.25

Units : 1/Year

```

利益 = 販売収入 - 総コスト
Units : 円/日

利益総額 = INTEG(利益,0)
Units : 円

受取 = DELAY MATERIAL(業者発注, 調達時間,0,0)
Units : 個/日

年日数 = 365
Units : 日/Year

店舗在庫 = INTEG(受取 - 販売実績, 注文 * (調達時間))
Units : 個

最大在庫 = 100
Units : 個

最小の業者発注 = IF THEN ELSE(店舗在庫 + 発注中<最小在庫,(最大在庫 -
    発注中 - 店舗在庫) / TIME STEP,0)
Units : 個/日

最小在庫 = 20
Units : 個

業者の予測 = IF THEN ELSE(補充日<>0,(注文予測 * (調達時間 +
    補充日数) - 店舗在庫 - 発注中) / 補充日,0)
Units : 個/日

業者発注 = IF THEN ELSE(発注ルールのスウィッチ = 1, 業者の予測,
    IF THEN ELSE(発注ルールのスウィッチ = 2, 最小の業者発注,
        別の発注量))
Units : 個/日

注文
Units : **undefined**

注文予測 = INTEG((注文-注文予測) * 重み付け, 注文)
Units : 個/日

発注コスト = IF THEN ELSE(業者発注 > 0, 処理コスト + 輸送コスト,0)
Units : 円/日

発注ルールのスウィッチ = 2
Units : Dmml

発注中 = INTEG(業者発注-受取,0)
Units : 個

発注単位 = 1
Units : 個/日

総コスト = 販売コスト + 倉庫コスト + 発注コスト
Units : 円/日

総利益 = INTEG (利益,0)
Units : 円

補充日 = IF THEN ELSE(MODULO(Time, 補充日数) = 0,1,0)
Units : 日

補充日数 = 10
Units : 日

調達時間 = 5

```

```

Units : 日

販売コスト = コスト単価 * 販売実績
Units : 円/日

販売単価 = 200
Units : 円/個

販売収入 = 販売実績 * 販売単価
Units : 円/日








販売実績 = IF THEN ELSE(店舗在庫-注文 > 0, 注文, 店舗在庫)
Units : 個/日

輸送コスト = 輸送割合 (業者発注 / 発注単位)
Units : 円/日

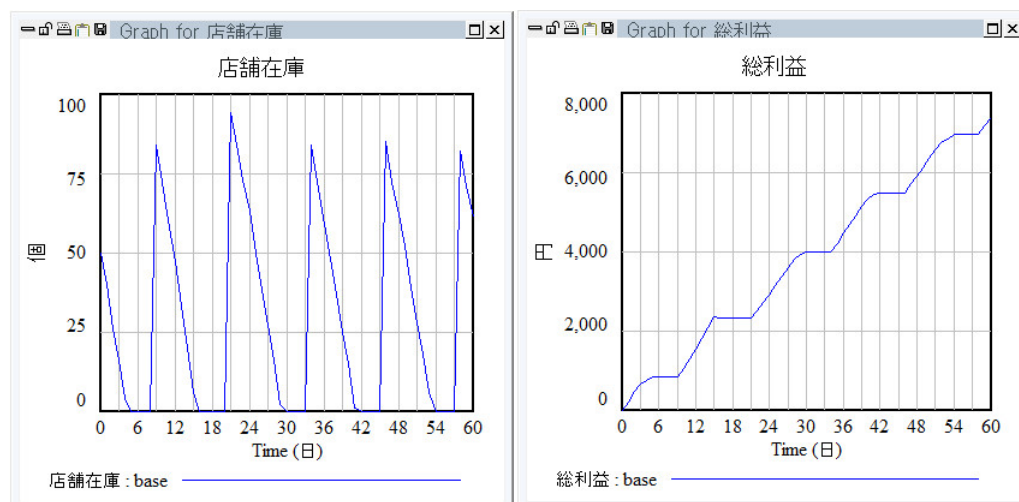
輸送割合 ([ (0,0)-(800,250) ], (0,10), (5,30), (15,75), (30,100), (50,125),
          (100,165), (150,180), (200,185), (700,235) )
Units : 円/日

重み付け = 0.2
Units : 1/日

```

- [モデル]メニュー [データセット読込]を選択し、*ORDERSjap.dat* を選択し、[開く]をクリックします。
- [Conversion completed without error]【エラーなしに変換しました】のメッセージが表示されたら [OK] をクリックし、出力ウィンドウを閉じます。
- シミュレーション名の編集ボックス  をダブルクリックし、「base」と入力します。
- シミュレーションの準備  をクリックします。
- [Euler] の右側のデータセット編集ボックスをクリックし、*ORDERSjap* と入力します。
- シミュレーションの実行  をクリックします。
- 制御パネル  をクリックし、[データセット]タブをクリックします。データセット *ORDERSjap* をダブルクリックし、除外します。
- 「店舗在庫」をダブルクリックし、グラフ  をクリックします。
- 「総利益」をダブルクリックし、グラフ  をクリックします。
- 出力ウィンドウ表示/入れ替え  をクリックし、背面のグラフを前面に表示させます。





英語データの *ORDERS.dat* の変数名を日本語に変更しています。



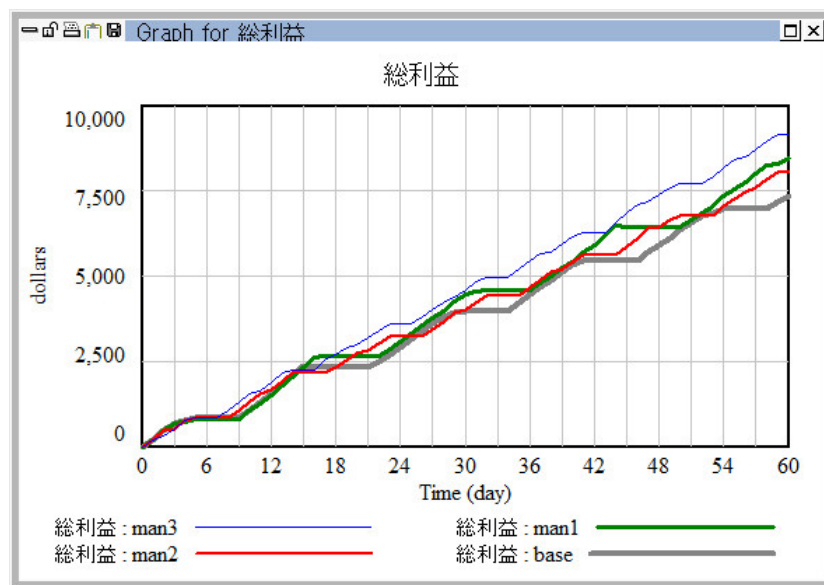
このモデルでは、在庫が 20 個（在庫最小値）まで下がると、100 個（在庫の最大値）まで増やします。しかし、補充される前に、「在庫」が何度かゼロになることに注目してください。

### 18.3.2 手動の最適化


練習のために、手動でモデルを最適化してみます。

- シミュレーションの準備  をクリックします。シミュレーション名の編集ボックス  をクリックし、「manual」と入力します。
- 「最小在庫値」と「最大在庫値」をクリックし、値を変更します。
- シミュレーションの実行  をクリックします。
- 「総利益」を選択し、ワークベンチ変数とし、グラフ  をクリックします。

このシミュレーションでの「総利益」の最後の値を、base と比較します。以下のような結果を得ることができます。最大の「総利益」になるように、試行錯誤を続けます。



### 18.3.3 ペイオフの設定

- シミュレーション名の編集ボックス  をダブルクリックし、「optimize」と入力します。
- 最適化の開始  をクリックします。ペイオフのダイアログが開きます。
- [タイプ]の[Policy]【方策】のラジオボタンをクリックします。[変数]の右側の[選択]をクリックし、リストから「在庫」を選択し、[OK]をクリックします（もしくは、編集ボックスに直接入力します）。
- [ウェイト]編集ボックスをクリックし、「1」と入力します。[編集して追加]をクリックします。

正のウェイトはそのパラメータが増えるほど良い（利益が上がる）ことを意味します。反対に負のウェイトは、増えると良くないパラメータです。ペイオフに2つ以上のパラメータを設定する場合は、パラメータのバランスを保つ必要があります。ペイオフの設定については、この章の初めの補正の例もしくは、リファレンスマニュアルの10章を参照してください。ペイオフダイアログは以下のようになります。

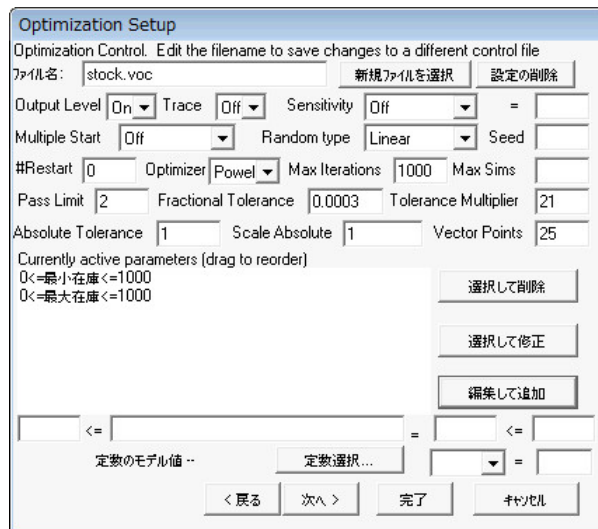
➤ [ 次へ ] をクリックします。

方策の最適化は、ペイオフの積分を最大にします。したがって、ペイオフで「利益」を使用することは、結果的にシミュレーションの最後に「総利益ペイオフ」を最大にするのと同じことです。

#### 18.3.4 最適化パラメータの設定

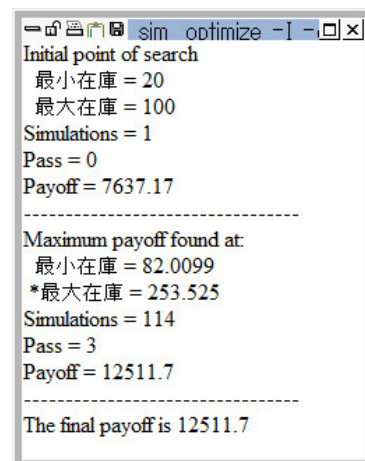
- [ 定数選択 ] をクリックし、リストから [ 最小在庫 ] を選択します。最小値 ( = の左側のボックス ) に「0」と入力し、最大値 ( = の右側のボックス ) に「1000」と入力します。[ 編集して追加 ] をクリックします。
- [ 定数選択 ] をクリックし、リストから [ 最大在庫 ] を選択します。最小値 ( = の左側のボックス ) に「10」と入力し、最大値 ( = の右側のボックス ) に「1000」と入力します。[ 編集して追加 ] をクリックします。




最適化の設定は以下のようになります。



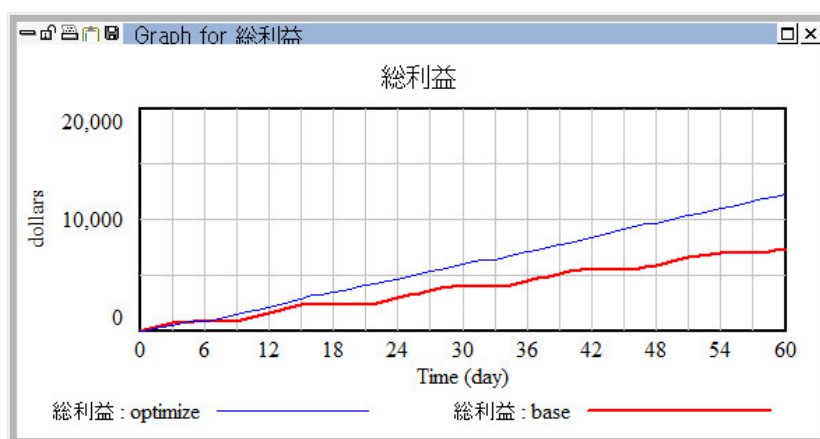
➤ [完了] をクリックします。

最適化が実行され、出力ファイルを生成し、最適値の実行結果を読み込みます。



- 制御パネル  をクリックし、[データセット] タブをクリックします。「base」と「optimize」以外のすべてのデータセットを、取り除きます。
- 「総利益」を選択し、ワークベンチ変数とします。グラフ  をクリックし、テーブル  をクリックします。

「総利益」の最後の値を、「base」と「optimize」で比較します。



Time (day)	55	56	57	58	59	60
"総利益" Runs:	optimize	base				
総利益	11,190	11,410	11,632	11,854	12,059	12,244
: base	6,963	6,963	6,963	6,963	7,173	7,364

グラフは、最適値での振る舞いを示しています。テーブルは、最適化によって得られた「総利益」の最大値を示します。



## 第 19 章

# モデルの配布

作成したモデルを、他の人へ簡単に送ることができます。Vensim を持っていない人でも、Vensim Model Reader を使って、モデルを表示したり、シミュレーションすることができます。Vensim の下位のラインアップでも、多くのモデルを読み込めます。したがって、Vensim DSS で開発したものであっても、Vensim PLE で操作できます。

Vensim DSS を使用している場合、モデルのパッケージ化や出版のための追加オプションがあります。これらは、DSS リファレンスマニュアルの 13 章に詳しく紹介されています。

### 19.1 Vensim Model Reader

Model Reader は、Vensim で作成されたモデルの読み込み専用のアプリケーションです。他の人が、Vensim を使用しなくてもシミュレーションや分析ができるように、モデルと一緒に、無料の Model Reader のインストールプログラムを配布できます。Vensim の CD を持っている場合、Model Reader のインストーラーは、*windows* ディレクトリの中、*venred32.exe* という名前のファイルです。ウェブサイトからダウンロードすることもできます。

Model Reader は、Vensim PLE から DSS までのラインナップで作成されたモデルを操作できます。Vensim DSS で作成したモデルは、外部関数を利用して作成されたものを除いて、Model Reader で操作できます。Model Reader を使用して、DDE 接続を行うことはできません。DDE 接続は、Vensim DSS のみ対応します。

Model Reader には、Venapp を用いる場合を除いて、感度シミュレーションもしくは最適化の機能はありません。また、Vensim Professional と DSS が、シミュレーションコントロールダイアログを通して利用できる他のシミュレーションの結果を再利用する機能はありません。しかし、ツールバーでデータソースを明示し、Vensim Model Reader の [ 変更 ] メニューからファイルを変更できます。

統合シミュレーションは Vensim Model Reader で操作できますが、変数の振る舞いに優先機能を持たせることはできません。

DDE とはダイナミックデータエクステンジのこと。ソフト同士がデータをやりとりする手順の 1 つです。

Venapp は、DSS で開発されたアプリケーションのことです。

### 19.1.1 モデルの下準備

Vensim Model Reader だけを持つ人に対して、モデルが利用できるようにするには、2つの方法があります。1 つは、[ ファイル ] [ 出版 ] を使用して、モデルのパッケージとサポートファイルを作成することです。もう 1 つは、バイナリ形式でモデルを保存することです。これらの準備とは別に、配布先でのモデル操作をより簡単にするために有効なことがいくつかあります。

まず、モデル作者の名前と連絡先を含めることがあります。最初の表示画面に、この情報をコメントとして挿入できます。またこうした表示画面を使う方法以外に、モデルノート([ モデル ] [ 設定 ] から [ 情報/パスワード ] タブをクリックします)を使用できます。このダイアログに情報を書き込み、[ モデルを開いたとき注意を表示 ] のチェックボックスにチェックを入れます。この情報は、モデルを開いた時にダイアログで表示されます。

各変数に、測定の単位とコメントを入力することも有効です。これらは、モデル操作中にマウスを変数上で動かすことによって表示され、よりモデルを理解しやすくします。

モデルが 2 つ以上の表示画面を持つ場合、最初のページに、ナビゲーションページを加えることが有効です。ナビゲーションページには、作者の連絡先や、モデルの各表示画面へ移動するための、ナビゲーションボタンを置くことができます。Vensim Model Reader では、ページアップ [PgUp] やページダウン [PgDn] の操作ができますが、ナビゲーションボタンは、これをさらに容易にします。この場合、それぞれの表示画面上には、最初の表示画面に戻るナビゲーションボタンを置いた方が良いでしょう。

ユーザーに変更してほしいと思う重要な定数や、重要な結果などは、制御パネルを用いてカスタムグラフを作成すると便利です。この例が 12 章にあります。

#### カスタムグラフと表

これは、Vensim PLE と PLE Plus は対応しません。

別のファイルにカスタムグラフもしくはテーブルを作成した場合、配布先にもその情報を渡す方法があります。これを行うために、制御パネルの [ グラフ ] タブをクリックします。

このダイアログは、  
*model\sample\EXTRA\*  
*WORLD.mdl* での表示  
です。



[ 自作グラフセット ] の下に表示されている名前が、[ \*Default ] でない場合、[ Into Model ] ボタンが選択可能ならば、カスタムグラフとテーブルを取り込むことができます。

それによって、現在のモデルにカスタムグラフや表の設定を読み込むことができます。

### 19.1.2 パッケージモデルの出版

Vensim Model Reader は、パッケージモデル (.vpm) のファイルと、バイナリ形式のモデル (.vmf) だけを読み込みます。これらはどちらも、Vensim だけで読み込めるバイナリファイル形式です。

注意 Vensim5.5 以前のバージョンでは、パッケージモデルは開くことができません。

### 19.1.3 パッケージの出版

モデルを出版するために、モデルが操作できる状態にある時、[ ファイル ] [ 出版 ] を使用します。モデルパッケージダイアログが表示されます。



名称 作成されるパッケージの名前と同じです。デフォルトで、読み込んでいるモデルのファイル名に、拡張子.vpm を加えたパッケージモデルが作成されます。Venapp (DSS のみで作成できます) をパッケージしている場合は、Vensim パッケージアプリケーションとなる、拡張子.vpa のファイルになります。

Also Include パッケージに含めたい他の要素を示します。使用している Vensim のラインアップなどによって、選択できない (グレー表示になる) ものがあります。

- [ Setting ]: チェックされている場合、ウィンドウズディレクトリもしくは、ユーザーのディレクトリにある、拡張子が.ini のファイルを含みます。これは、問題を解決するためにパッケージを送る場合は有効ですが、そうでなければ使用しません。
- [ Toolset ]: チェックされている場合、現在読み込まれている分析ツールセットを含みます (保有されている分析ツールセットが読み込まれている場合のみ利用できます。Vensim PLE および PLE Plus では、常にグレー表示されています)。
- [ vgd.file ]: チェックされている場合、保存されている Vensim グラフ定義ファ

イル（デフォルトを除く）を含みます。Vensim PLE および PLE Plus では、常にグレー表示されています。

- [Extern dll]: チェックされている場合、利用されている外部関数のライブラリーを含みます。外部関数のライブラリーを読み込んで Vensim DSS を使用している場合を除いて、これはグレー表示されています。特にライブラリーを再配布できるかどうかを確認してください。

上記のファイルは、コンピューター上の異なる場所にあったとしても、同じディレクトリの中にあるパッケージとしてモデルが保存されます。Extern DLL だけが、パッケージで用いられる際に、引き出され、適切なディレクトリに配置されます。

**Additional Files** 含むべき追加のファイルをリストにします。このダイアログが開いた時、モデルに関連したすべてのファイル（読み込まれたデータセット、更新履歴、データファイル、最適化コントロールファイルなど）をリストに加えることができます。リストに何か追加する場合は、[追加] をクリックします（Windows では **[Shift]** + クリックもしくは、**[Ctrl]** + クリックで、複数のファイルを選択することもできます）。リストから何か削除したい場合は、それを強調表示し、[除去] をクリックします。Excel ファイルをデータソースとして使用している場合、自動的にリストに追加されないことに注意してください。

もし、データセット（.vdf）を含めてパッケージを作成した場合、これらのファイルは読み込み専用ファイルとなります。したがって、データセットと同じ名前でシミュレーションしようとした場合、エラーメッセージが表示され、異なるファイル名を作成するように求められます。

**Package Type** アーカイブ形式を指定します。これは、Vensim DSS のみ対応します。

[Model] は、モデルおよびサポートファイルを含みます。[Venapp] は、Venapp ファイル（.vcd）およびサポートファイル（モデルを含みます）を含みます。アーカイブを開くと、Venapp が実行されます。

**Readable by** どのアプリケーションでパッケージモデルを開くか管理します。例えば、モデルへのアクセスを制限したい場合、Model Reader のみ読み込み可能とします。

[Model Reader] は、Vensim Model Reader を指します。[PLE/PLUS/Std/Pro] は、Vensim PLE、Vensim PLE Plus、Vensim Standard、Vensim Professional を指します。[DSS] は、Vensim DSS を指します。[DLL] は、Vensim DLL を指します。これは、Vensim DLL のみ対応します。[Runtime] は、Vensim Runtime を指します。[Application Runtime] は、Vensim Application Runtime を指します。これは Venapp の Package Type が選択された、Vensim DSS のみ対応します。[Special] は、Vensim エンジンを使用する Sable のようなアプリケーションを指します。

**No Save As** チェックされている場合、モデルを異なる名前で保存したり、変更を加えられないようにします。これは、Vensim Model Reader と同じ機能を、すべての Vensim アプリケーションに持たせます。

**No Copying** チェックされている場合、モデルの構造をコピーすることを防ぎます。これは [No Save As] より強力です。つまり、表示画面でモデル構造を見たり、ドキュメントツールを使用したりできますが、[編集] [コピー] はできません。

**Scramble binaries for redistribution** チェックされている場合、許可された再配分可能な

Vensim を使用するアーカイブを作成します。再配送ライセンスを持っている場合のみ、このオプションは利用できます。

Password Protect チェックされている場合、モデルを開くためにパスワードが必要になります。使用したい [ パスワード ] を入力し、[ 認証 ] してください。パスワードをなくした場合、モデルは開けません。

注意 パスワードによる保護は、モデルの興味本意なアクセスを防ぎます。しかし、それを見たいという欲望を止めることは難しいです。知的財産の安全性と保護は、重要な問題であり、自己責任の部分があります。

Other details for problem solving Vensim とコンピューターについての情報を表示します。これは、モデルにおける問題解決のためのサポートを受ける場合に有効です。追加したいコメントをここに書き込むこともできます。

Setting5: Load... 以前に保存されたこのダイアログの設定を読み込みます。これは、一度作成した出版の設定を保存し、それを再利用することを可能にします。設定を読み込む時、同じモデルのパッケージである必要はありません。しかし、それらは同じディレクトリの中にある必要があります。

Settings: Save... 現在のダイアログの設定を保存します。これは、この設定を再利用するために用います。パスワードを設定している場合、保存するかどうか質問されます。その場合、パスワードはフォームファイル (.frm) にプレーンテキストで保存されます。フォームファイルは、パッケージモデルに含まれないので、パスワードが知られることはありません。

モデルの出版する設定によって、それを開く Vensim のラインアップが決まります。場合によっては、使用している Vensim のラインアップで利用できない機能を持つかもしれません。また、Vensim DSS および Vensim Model Reader だけがパッケージアプリケーションを開くことができます。

#### 19.1.4 バイナリ形式

モデルをバイナリ形式で保存するために、[ ファイル ] [ 別名で保存 ] を使用し、[ Binary Format Models ] を選択し、新しいファイルの名前を入力します。この名前に追加される拡張子は .vmf です。[ ファイル ] [ 別名で保存 ] のダイアログで、拡張子を .mdl から .vmf に変更するだけで、モデルをバイナリ形式のモデルとして保存することもできます。

Vensim のどのラインアップでも、.vmf ファイルを開くことができます。これらは、純粋なテキスト形式のファイル.mdl よりも大きく、Vensim だけで開くことができます。テキストエディタによって、.mdl ファイルを開くことができるので、テキスト形式はバックアップなどにおいて有効です。

## 19.2 パッケージモデルを開く

Vensim のモデルを開くのと同様に、パッケージモデルを開きます。パッケージモデルは [ ファイル ] [ 開く ] を使用した時に表示されます。また、Windows のエクスプローラによって、パッケージをダブルクリックすることもできます。しかし、パッケー

ジモデルの外観は少し異なります。

モデルが別名で保存できないよう設定されている場合、スケッチツールはありません。  
[ファイル] [保存]と[表示] [テキスト表示]は操作できません。

別名で保存ができる設定ならば、スケッチツールが表示され、ファイルを別名で保存するダイアログが開きます。モデルに変更を加えることができますが、変更後のファイルは、パッケージとは別に保存しなければなりません。パッケージそのものは、読み込み専用です。

パッケージがデータセットを含んでいる場合、そのデータセット名は使用できません。制御パネルから、データセットを読み込むことも取り除くこともできます。しかし、オペレーティングシステム上ではファイルとして表示されません。

パッケージがツールセットを含んでいる場合、Vensim Standard、Professional もしくは DSS は、そのツールセットを読み込むかどうか質問されます。ツールセットが自動的に読み込まれる Vensim Model Reader では、どんな埋め込まれたグラフも予想通りに現れることを保証します。Vensim PLE と PLE Plus では、これらのラインアップがツールセットの修正をサポートしていないので、ツールセットは無視されます。

パッケージがシミュレーションの実行のためのウィザードを使用した感度分析や、もしくは最適化コントロールファイルを含むことがあります。それを実行すると、パッケージから読み込まれ、変更した設定をファイルに保存します。しかし、パッケージを再び開くと、パッケージコントロールファイルから開始します。パッケージは読み込み専用であり、開いた時に常に同じ結果が得られるよう作られています。

パッケージモデルを開き、他の名前で保存した場合、パッケージの中に含まれたデータセットは、まだ利用できます。しかし、新たに作成したモデルを開いて改めて開くと、前のデータセットは利用できなくなります。

一度に 1 つのパッケージモデルしか開くことはできません。2 つめのモデルを開くと、最初のパッケージに関連したモデルが閉じます。また、パッケージモデルもしくは、パッケージに含まれたファイルを出版することはできません。

パッケージモデルにあるファイルと同じ名前を持つファイルがある場合、パッケージモデルにあるファイルが使用されます。混乱を避けるために、多くのファイルがあるディレクトリにパッケージモデルを置かない方が良いです。

### 19.3 Vensim PLE によるモデルの利用

修正できるモデルを提供したいこともあります。Vensim PLE は他の Vensim のラインアップで開発されたモデルを開くことができます。しかしながら、すべての Vensim の機能が Vensim PLE に対応するわけではないため、すべてのモデルが操作できるわけではありません。

**下添え字** 下添え字を使用したモデルは、Vensim PLE では、エラーメッセージを表示します。モデルの構造は見ることはできますが、シミュレーションできません。

**サポートされない関数** サポートされない関数を使用したモデルは、Vensim PLE で開くことはできますが、シミュレーションできません。[モデル] [モデルチェック]を使用することで、どの関数がサポートされていないか表示できます。

ナビゲーションコメント ナビゲーションコメントは、Vensim PLE で表示できますが、ナビゲートはできません。

## 19.4 E メールとウェブ利用上の注意

.mdl 形式のモデルは、プレーンテキストファイルです。これは、モデルを管理しやすくする一方、配布時に、予期しない問題をもたらします。これらのモデルを E メールで送るか、もしくはウェブサイトに置く際に、知っておくべきことがあります。

いくつかの E メールプログラムは自動的にテキスト文の行を結合します。こうした文の変形が変数名やスケッチの定義をまたぐと、モデルが壊れます。この典型的な兆候は、syntax エラーメッセージか、もしくは不完全なスケッチのどちらかです。テキストエディタ（例えばノートパッド）でファイルを開いた場合、どこで改行がされているかを見ることができます。この種の問題を防ぐには、これらを送る前に、zip ファイルにしておくことです。

ウェブサイトに置かれたテキストファイルには、行間に、しばしば空白の行が入っています。これは、Unix と Windows 及び Macintosh のファイルシステム間で、テキストファイル形式に互換性がないためです。これが問題かどうか確認するために、ウェブサイトに置いたファイルをチェックします。E メールで添付されたテキストを、ウェブブラウザに開いた場合にも、よく似た問題が起こります。うまく働くかどうか確認するために、添付ファイルを保存します。

これらの問題はどちらも、モデルの出版によりパッケージモデルを作成することで防ぐことができます。






## 第 20 章

# 参照モード


参照モードは、設定された課題に対して、重要変数の時間による変化を表すグラフです。例えば、課題設定が、商品の価格が周期的に上下する場合には、参照モードは価格のグラフになります。これは、これまでのデータ収集による推定、もしくは正弦波を用いた理想モデルなどを用います。さらに参照モードは、起こった出来事を表すことに加えて、この先起こる出来事に対しても利用できます。例えば、市場に供給する新しい製品を作り出した場合、そうあってほしいと望む継続した成長パターンと、成長後に崩壊するパターンの参照モードがありえます。

参照モードは、課題設定を明確にするために役立ちます。参照モードを描くためには、調べるべき変数の変化をみる期間を特定する必要があります。このことは、モデルの変数に何を含み、何を除くかを決定するにあたって大変有効です。参照モードは、モデルにおいて、どの変数が最も重要かを見分けるためにも役立ちます。もし変数に明瞭な参照モードがなければ、その変数はモデルには不要かもしれません。参照モードはまた、主要な問題ではない振る舞いの要素を除くことによる要約に対して有効です。例えば、ビジネスサイクルのダイナミックスを見ている場合に、成長のダイナミックスは除外することなどがあります。こうした要約は、モデルをわかり易くする上で、大変役立ちます。

この章では、参照モード  により、モデルへ参照モードを加える方法を紹介します。また参照モードとシミュレーションによる振る舞いを比較します。また参照モードにおける外生データの利用方法も紹介します。なぜなら外生データを入力し、参照モードで想定される振る舞いを示すことは、メンタルモデルを検討する上で有効なためです。

### 20.1 参照モードによる概念化


参照モードは、モデルを概念化し、他からの情報を得る上で役立ちます。ここでは、簡単な例を用いて紹介します。工場の整備を担当する作業員が減少する中での、工場稼働の問題を考えます。ここでの想定は、減少した作業員は、機械の整備に多くを要するため、思った以上の影響が生じるということです。この問題を理解するために重要な変数を作成します。

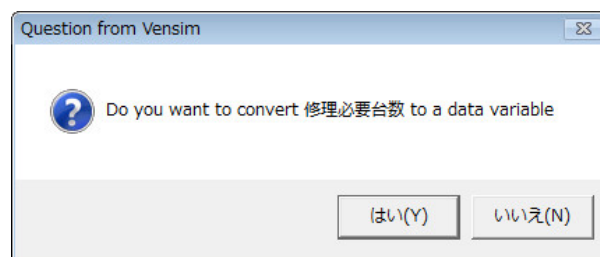
- [ファイル] [新規モデル] を選択し、新しいモデルを作成します（デフォルトの時間の範囲を使用します）。
- 変数  をクリックし、「修理必要台数」と「修理担当者数」を加えます。以下の

ようになります。

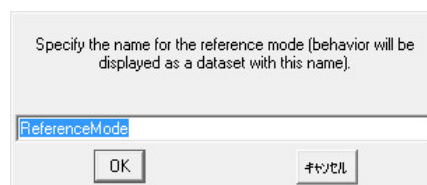
修理必要台数

修理担当者数

- 参照モード  を選択し、「修理必要台数」をクリックします。
- [ Do you want to convert 修理必要台数 to a data variable ] [「修理必要台数」をデータ変数に変換しますか] と表示されるので、[ いいえ ] をクリックします（これは、後で説明します）。



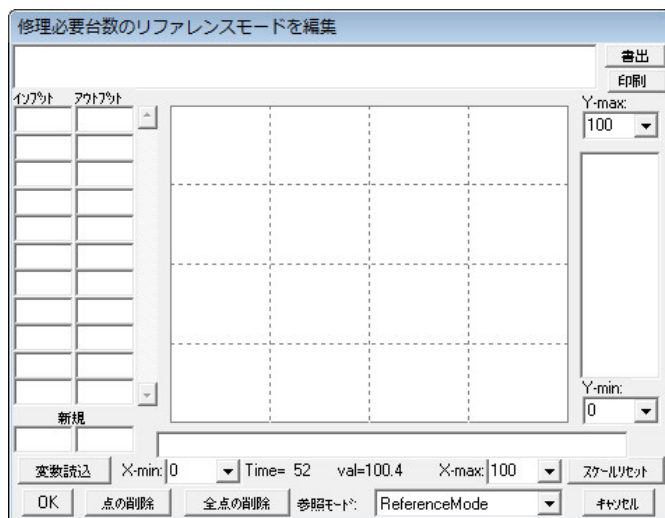
参照モードの名前を質問されます。



デフォルトの名前 “ReferenceMode” がよいでしょう。異なる名前を選択することもできます。また、参照モードは 8 個まで作成することもできます。

- 参照モードダイアログで [ OK ] をクリックします。

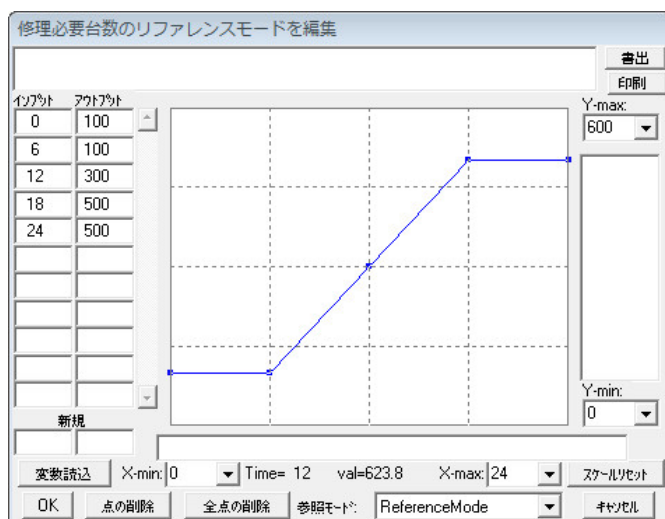
表関数編集とほとんど同じダイアログボックスが開きます。





表関数編集ダイアログとは異なり、x 軸は時間軸とみなされ、下にはドロップダウンボックス（ ）があります（[ 参照モード: ] と表示されています）。ここでは、異なる参照モード（8 つまで定義できます）を選択することや、新しい参照モードを作ることが可能です。この例では、1 つの参照モードだけを使用します。

時間軸は 0 から 100 になっています。これは、ここでの問題に適切ではないため、24 カ月に変更します。

- [ キャンセル ] をクリックします。
- [ モデル ] [ 設定 ] を選択し、モデル設定ダイアログで、[ 終了時間 ] を「24」に変更します。[ OK ] をクリックします。
- 「修理必要台数」を再度クリックし、参照モード編集ダイアログを開きます（データ変数への変換には、再度 [ いいえ ] をクリックします）。
- [ インプット ] と [ アウトプット ] の欄にそれぞれ「0,100」「6,100」「12,300」「18,500」「24,500」と入力します。ダイアログは以下ようになります。



- [ OK ] をクリックし、ダイアログを閉じます。
- 「修理担当者数」をクリックします。
- これをデータ変数にするため、[ はい ] をクリックします。

データ変数は時間とともに変化する入力値です。データ変数は外生変数とも呼ばれます。「修理担当者数」をデータ変数に変換した場合、これは内部で計算されないで、データファイルを読み込むことを意味します。シミュレーションのためのデータファイルを指定していない場合、最初に作成した参照モードファイルがデータファイルとなります。シミュレーションの準備  からこれを変更することができます。また、制御パネル  の [ advanced ] タブからも変更できます。「修理担当者数」をデータ変数にするのは、その変更によって起こることを理解するためのモデルであるからです。

- 「修理担当者数」は 10 人で開始します。しかし 6 か月を過ぎると半分にカットされます。これを示すために、「0,10」「6,5」「24,5」と入力します。
- [ OK ] をクリックし、ダイアログを閉じます。

モデルは以下ようになります。

修理必要台数

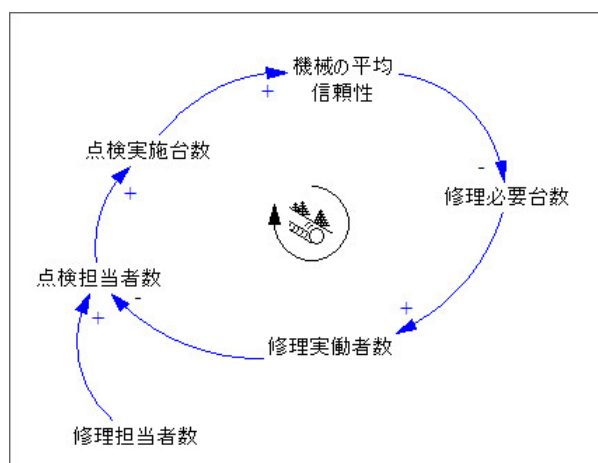
修理担当者数


### 20.1.1 ループの作成

入力と結果に対応した変数を作成しました。この 2 変数は異なる参照モードを持っています。それらが妥当なものならば、これらの変数を関連付ける構造が必要です。そこで変数を追加して、ダイナミックな構造を作成します。

英語版です。

- 以下のような図を完成させるか、もしくは `guide\chap20\complete\repair.mdl` を開きます。

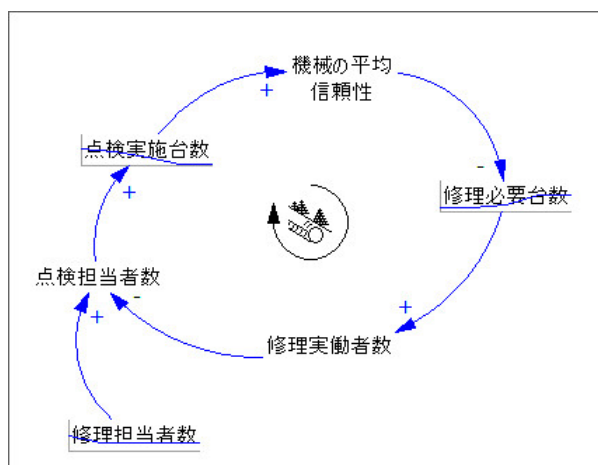


➤ 参照モード  を選択し、「点検実施台数」をクリックします。

今回は、「点検実施台数」をデータ変数にするかどうか質問されません。なぜなら、それが内生的に計算されることが明らかのためです。同じことは「修理必要台数」についても言えます。しかし、初めに参照モードを加えた時はそうではありませんでした。

➤ [インプット]と[アウトプット]の欄にそれぞれ「0,200」「6,175」「12,100」「18,0」「24,0」と入力し、[OK]をクリックします。

以下のような図になります。



これはダイナミックな仮説を表現し、参照モードはその内容を分かりやすくします。さらに、他の変数にも、参照モードを作成することもできます。これは、どんな振る舞いになるかを理解する上で、よい練習になります。参照モードはモデルによって生成されているのではなく、モデルの製作者によって作られているということを忘れないでください。



このモデルのすべての方程式は、以下のようになります。

平均故障率 = 整備機械の故障率 \* 機械の耐久効果  
Units : 1/月

点検・修理後の平均時間 = 点検・修理後の時間/機械台数  
Units : 月

機械の耐久効果 = WITH LOOKUP (点検・修理後の平均時間/機械の耐久時間,  
([ (0,0)-(4,8) ], (0,0.9), (1,1), (1.5,1.1), (2,1.5),  
(3,3), (4,6) ))  
Units : Dmnl

機械の耐久時間 = 2  
Units : 月

整備機械の故障率 = 0.1  
Units : 1/月

整備の生産性 = 120  
Units : 台/人/月

修理必要台数 = 平均故障率 \* 機械台数  
Units : 台/月

修理完了 = 修理実働者数 \* 修理能力  
Units : 台/月

修理能力 = 30  
Units : 台/人/月

点検実施台数 = MIN(機械台数/機械の耐久時間, 点検担当者数 \* 整備の生産性)  
Units : 台/月

機械台数 = 1000  
Units : 台

修理担当者数  
Units: 人

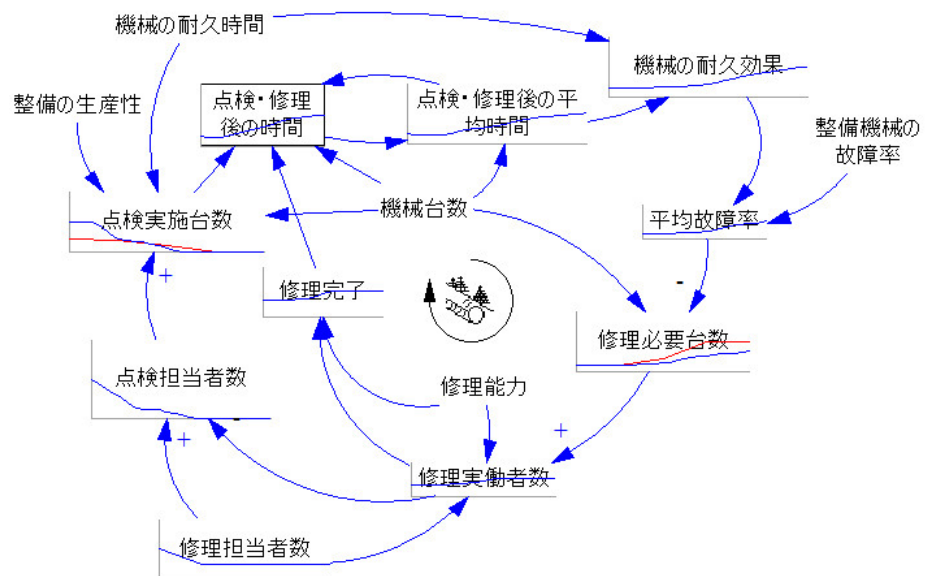
点検・修理後の時間 = INTEG (機械台数 - (修理完了 + 点検実施台数) \*  
点検・修理後の平均時間, 機械台数 \* 機械の耐久時間)  
Units : 台\*月

点検担当者数 = 修理担当者数 - 修理実働者数  
Units : 人

修理実働者数 = MIN(修理必要台数/修理能力, 修理担当者数)  
Units : 人

「修理担当者数」の方程式が空白になっています。これは、データ変数を表しています。もしこの章の初めの作業を行っていれば、方程式を追加する必要はないでしょう。なぜならデータ変数が読み込まれるからです。しかし、単位は追加する必要があります。

このモデルをシミュレーションし、その振る舞いを表示した場合、( **b** )は振る舞いの表示のオン・オフを切り替えます) 以下のようになります。



部分的に複製した 2 つの参照モードを用いました。その振る舞いはシミュレーションと同じというよりも、似ている程度です。参照モードとシミュレーションによる振る舞いに、いくつかの違いがあることは、一般的なことです。

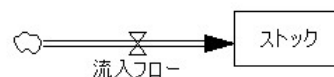
参照モードを用いて、概念化からモデルを開発し、シミュレーションを通じてモデルを完成させました。参照モードは初期のモデル開発の過程で振る舞いの具体的な表現を追加します。また、外生データを分かり易く示します。


### 20.1.3 メンタルの統合

参照モード編集ツールは、メンタルの統合技術として有効な手段です。積分や蓄積はダイナミックな振る舞いを生み出す基礎です。また、状況に応じた蓄積過程を考えることは、ダイナミックな問題を理解する上で有効です。

- デフォルトの時間範囲を使用して、以下のようなモデルを作成するか、もしくは `guide\chap20\complete\MentalSimulation01.mdl` を開きます。


英語版です。

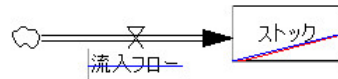



- 参照モード  をクリックし、「流入フロー」をクリックします。これをデータ変数に変えるかどうかの質問に、「はい」と答えます。
- 参照モードの名前を「MentalSimulation1」とし、「0,10」「100,10」と入力します。
- [OK] をクリックし、ダイアログを閉じます。

この「流入フロー」で、ストックの値がどうなるかを予想できます。



- 方程式  $f(x)$  を選択し、ストックの初期値として「0」を入力します。
- 参照モード  を選択して「ストック」をクリックし、予想値を入力します。
- シミュレーションを実行します。以下ようになります。




- 参照モード  を選択し、「流入フロー」をもう一度クリックします。「0,0」「50,100」「100,0」と入力します。
- 「ストック」をクリックし、予想する値を入力します。シミュレーションを実行し、予想と比較してください。
- 次のような異なる「流入フロー」のパターンで、この種の実験を繰り返します。例えば、「0,0」「100,100」や「0,100」「100,0」や「0,100」「50,0」「100,100」とします。

「流入フロー」が1つだけでも、メンタルなシミュレーションは驚くほど巧妙です。ここに2つ目のフローが追加されると、問題はさらに難しくなります。

- 以下のように流出フローをモデルに追加するか、もしくは `guide\chap20\complete\MentalSimulation02.mdl` を開きます。

英語版です。



- 参照モード  を選択し、「流出フロー」をクリックします。これをデータ変数に変えるかどうかの質問に、「はい」と答えます。「流出フロー」のパターンを「0,0」「50,100」「100,0」と設定します。
- 「流入フロー」のパターンを「0,100」「50,0」「100,100」と設定します。
- 方程式  $f(x)$  を選択し、「ストック」の方程式を次のように変更します

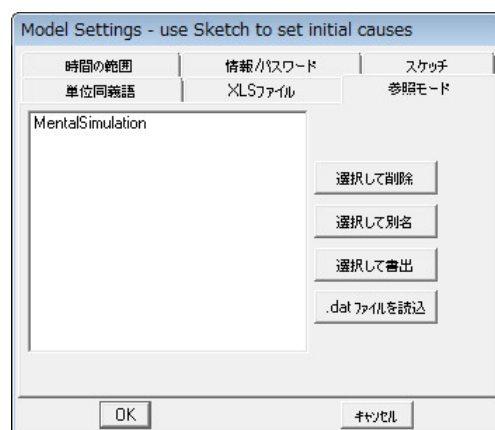
ストック = INTEG(流入フロー - 流出フロー, 0)

- 「ストック」に対して予想する値を入力します。

簡単な入力パターンでも、「流入フロー」と「流出フロー」から影響を受けるストックの振る舞いは、予想できないものになります。「流入フロー」と「流出フロー」に、より複雑なパターンを入力し、予想通りの結果が得られるか試します。

#### 20.1.4 参照モードの管理

モデルに、8つまでの参照モードを設定することができます。[モデル] [設定]を選択し、モデル設定のダイアログの[参照モード]タブから、これらを管理できます。



変更したい参照モードをクリックします。

**選択して削除** 選択された参照モードをモデルから削除します。その参照モードで用いられたデータセットはそのままですが、制御パネルのデータセットタブから削除できます。

**選択して別名** 参照モードの名前を変更します。古い名前はデータセットとして残りますが、制御パネルのデータセットタブから削除することができます。

**選択して書出** 参照モードのインフォメーションを持つ .dat 形式のファイルを作成します。 .dat 形式のファイルの詳細は、リファレンスマニュアルの 9 章を参照してください。

**.dat ファイルを読み込** .dat 形式のファイルを読み込み、それを参照モードとすることができます。 .dat 形式のファイルの詳細は、リファレンスマニュアルの 9 章を参照してください。

#### 旧バージョンとの互換性

Vensim の 5.2 以前のバージョンでは、参照モードは対応していません。 Vensim の古いバージョンでも、参照モードを含んだモデルを開くことはできますが、参照モードの情報は消えています。これは、モデルを古い Vensim のバージョンで保存していなくてもそうなります。