



S⁴ Simulation System

Version 6.2

操作マニュアル



Copyright © 2008-2022 NTT DATA Mathematical Systems, Inc. All Rights Reserved.

目次

第 1 章	はじめに	13
1.1	シミュレーションについて	14
1.1.1	シミュレーションとは	14
1.1.2	シミュレーションのメリット	14
1.1.3	シミュレーションの分類	14
1.1.4	離散イベントシミュレーション	15
1.1.5	離散イベントシミュレーションの例	15
1.2	本システムの特徴	18
1.2.1	グラフィカルなモデル記述	18
1.2.2	部品編集によるカスタマイズ	18
1.2.3	より高度な部品のカスタマイズ	19
1.2.4	可視化機能	19
1.2.5	分析機能	20
1.2.6	多様な実行モード	21
1.2.7	スケーラブルで汎用的な psim 言語	22
1.2.8	ハイブリッドモデリング	22
1.2.9	エージェントシミュレーション	23
1.3	表記について	23
1.4	用語	23
第 2 章	起動と終了	28
2.1	起動	28
2.2	終了	28
第 3 章	ユーザインターフェース	30
3.1	ウィンドウ構成	30
3.1.1	メニュー、ツールバー	30
3.1.2	ブラウザパネル	30
3.1.3	情報パネル	33
3.1.4	モデル編集パネル	34
3.2	メニュー	34
3.2.1	ファイルメニュー	34
3.2.2	編集メニュー	36

3.2.3	モデルメニュー	37
3.2.4	ツールメニュー	38
3.2.5	ヘルプメニュー	38
3.3	ツールバー	39
3.3.1	作成・保存	39
3.3.2	インポート・エクスポート	39
3.3.3	元に戻す・やり直し	40
3.3.4	切り取り・貼り付け	40
3.3.5	モデル実行	40
3.3.6	サマリ表示	41
3.3.7	コード表示	41
3.3.8	モデル編集	41
3.4	ワークスペースタブのメニュー	42
3.4.1	ワークスペース	42
3.4.2	プロジェクト	43
3.4.3	モデル	44
3.4.4	入力セットフォルダ	44
3.4.5	出力セットフォルダ	44
3.4.6	入力フォルダ	45
3.4.7	出力フォルダ	46
3.4.8	モニター	47
3.4.9	時系列モニター	48
3.4.10	強化学習モデル	50
3.4.11	列	50
3.5	モデル編集のメニュー	51
3.5.1	モデル編集メニュー	51
3.5.2	部品・資源・フローアイテム・変数・エージェント・環境メニュー	53
3.5.3	リンクメニュー	54
3.6	メッセージタブ・システムタブのメニュー	55
3.6.1	クリア	56
3.6.2	コピー	56
3.6.3	全てを選択	56
3.7	モデル出力タブ・モデルエラータブのメニュー	56
3.7.1	クリア	56
3.7.2	コピー	56
3.7.3	全てを選択	56
3.7.4	最後のモデル出力(エラー)の結果を表示	56
3.8	Python エディタのメニュー	57

3.8.1	元に戻す	57
3.8.2	やり直す	57
3.8.3	切り取り	57
3.8.4	コピー	57
3.8.5	貼り付け	57
3.8.6	削除	57
3.8.7	全てを選択	57
3.8.8	外部エディタ	57
第 4 章	プロジェクト	58
4.1	基本操作	58
4.1.1	プロジェクトの作成	58
4.1.2	プロジェクトの削除	60
4.1.3	プロジェクトの名前の変更	60
4.2	プロジェクトのインポート・エクスポート	60
4.2.1	プロジェクトのインポート	60
4.2.2	プロジェクトのエクスポート	61
4.3	ワークスペース	61
4.3.1	ワークスペースの変更	61
4.4	データのインポート	61
4.4.1	ファイル指定	62
4.4.2	フォーマット指定	62
4.4.3	列選択	64
4.4.4	型・名前指定	66
4.4.5	時間列指定	67
4.4.6	データ名指定	69
4.5	データのエクスポート	69
4.5.1	ファイル指定	69
4.5.2	フォーマット指定	69
第 5 章	モデル	72
5.1	基本操作	72
5.1.1	モデルを開く	72
5.1.2	モデルの保存	72
5.1.3	モデルを閉じる	73
5.1.4	プロジェクトを別名で保存	73
5.2	元に戻す・やり直し	73
5.3	フローの編集	74
5.3.1	部品アイコンについて	74
5.3.2	部品の配置	75

5.3.3	リンクの作成	75
5.3.4	部品の選択	76
5.3.5	部品・リンクメニュー	77
5.3.6	選択部品の移動	77
5.3.7	選択部品の削除	77
5.3.8	リンクの削除	78
5.3.9	リンクを隠す/表示する	78
5.3.10	入力ポートの表示位置の変更	79
5.3.11	出力ポートの表示位置の変更	79
5.3.12	選択部品の切り取り	79
5.3.13	選択部品のコピー	79
5.3.14	選択部品の貼り付け	79
5.3.15	部品の名前の変更	79
5.4	パラメータ編集	80
5.4.1	基本設定タブ	80
5.4.2	パラメータタブ	83
5.4.3	一括実行タブ	84
5.4.4	感度分析タブ	88
5.4.5	実験計画タブ	94
5.4.6	最適化タブ	105
5.5	カスタムコード編集	111
5.6	分析スクリプト編集	111
5.6.1	スクリプト定義エディタ	112
5.6.2	データブラウザ	113
5.7	分析スクリプトの実行	114
5.8	コード表示	114
5.8.1	自動生成コードの構成	116
5.9	モデルの実行	118
5.9.1	通常実行	118
5.9.2	一括実行	120
5.9.3	感度分析実行	122
5.9.4	実験計画実行	124
5.9.5	最適化実行	126
5.10	実行結果	130
5.10.1	通常実行	130
5.10.2	一括実行	130
5.10.3	感度分析	131
5.10.4	実験計画	131
5.10.5	最適化	133

5.11	モデルのデバッグ	134
5.11.1	部品エラー	134
5.11.2	部品警告	134
5.11.3	その他のエラー	135
5.12	モデルプロパティ編集	136
5.13	ナビゲーションウィンドウ	136
5.14	Python 拡張環境を編集する	137
5.14.1	パッケージリスト	140
5.15	ソースコードをエクスポートする	140
5.16	ソースコードをインポートする	141
第 6 章	部品	143
6.1	部品の編集	143
6.1.1	共通設定タブ	143
6.1.2	属性設定タブ	146
6.1.3	コード編集タブ	147
6.2	生成 (Generate)	147
6.3	終端 (Terminate)	148
6.4	イベント待ち (WaitEvent)	149
6.5	イベント発火 (SignalEvent)	150
6.6	ファシリティ要求 (RequestFacility)	150
6.7	ファシリティ解放 (ReleaseFacility)	152
6.8	ファシリティ利用 (UseFacility)	152
6.9	ストアから取得 (GetFromStore)	153
6.10	ストアへ追加 (PutToStore)	155
6.11	タンクから取得 (GetFromTank)	157
6.12	タンクへ追加 (PutToTank)	158
6.13	結合 (Batch)	160
6.14	マージ (Merge)	161
6.15	分割 (Split)	162
6.16	設定 (Assign)	162
6.17	判断 (Decide)	163
6.18	遅延 (Delay)	164
6.19	複製 (Duplicate)	166
6.20	通過 (Pass)	167
6.21	記録 (Record)	167
6.22	連続 (Continuous)	168
6.23	即時報酬確定 (GetReward)	170
6.24	強化学習設定 (AssignRL)	171

第 7 章	リンク	173
7.1	ポートの動作	173
7.2	セレクタ	173
7.2.1	ランダム (一様)	173
7.2.2	ランダム (重み付き)	174
7.2.3	ラウンドロビン	174
7.2.4	固定	174
7.2.5	最短キュー	174
7.2.6	ユーザ定義	174
7.2.7	強化学習	175
7.3	接続方式	176
7.3.1	一对多 (出力キュー 0、入力キュー m)	176
7.3.2	一对多 (出力キュー 0、入力キュー 0)	177
7.3.3	一对多 (出力キュー n、入力キュー m)	177
7.3.4	一对多 (出力キュー n、入力キュー 0)	177
7.3.5	多対一 (出力キュー 0、入力キュー m)	179
7.3.6	多対一 (出力キュー 0、入力キュー 0)	179
7.3.7	多対一 (出力キュー n、入力キュー m)	180
7.3.8	多対一 (出力キュー n、入力キュー 0)	181
7.3.9	多対多	181
第 8 章	資源	182
8.1	資源の編集	182
8.1.1	共通設定タブ	182
8.1.2	属性設定タブ	183
8.2	イベント (Event)	183
8.3	ファシリティ (Facility)	184
8.4	ストア (Store)	185
8.5	タンク (Tank)	186
第 9 章	フローアイテム	188
9.1	フローアイテムの編集	188
9.1.1	共通設定タブ	188
9.1.2	属性設定タブ	189
9.2	アイテム (FlowItem)	190
第 10 章	変数	191
10.1	変数の編集	191
10.1.1	共通設定タブ	191
10.1.2	属性設定タブ	192

10.2	連続変数 (Value)	192
10.3	補助変数 (Expression)	193
第 11 章	エージェント	194
11.0.1	属性設定タブ	194
11.1	同期エージェント (SynchronousAgent)	195
11.2	SFM エージェント (SFMAgent)	198
11.3	粒子フィルタエージェント (ParticleFilterAgent)	200
11.4	NW エージェント (NWAgent)	202
11.5	非同期エージェント (AsynchronousAgent)	204
第 12 章	環境	206
12.0.1	属性設定タブ	206
12.1	SFM 地図 (SFMap)	207
12.2	SFM 環境 (SFEnvironment)	208
12.3	PowerlawCluster グラフ (PowerlawClusterRandomGraph)	210
12.4	NW 地図 (NWMap)	212
12.5	格子グラフ (LatticeGraph)	213
12.6	GraphML フォーマットグラフ (GraphMLFormatGraph)	215
12.7	GNP グラフ (GNPRandomGraph)	216
12.8	GNM グラフ (GNMRandomGraph)	218
12.9	GEXF フォーマットグラフ (GEXFFormatGraph)	220
12.10	ユークリッド 2D (Euclid2D)	222
12.11	環境 (Environment)	223
12.12	カスタムグラフ (CustomGraph)	225
12.13	完全グラフ (CompleteGraph)	226
12.14	BarabasiAlbert グラフ (BarabasiAlbertRandomGraph)	228
第 13 章	SFM 地図エディタ	230
13.1	SFM 地図エディタの概要	230
13.2	移動可能領域操作モード	231
13.2.1	領域選択ツール	232
13.2.2	矩形作成ツール	233
13.2.3	多角形作成ツール	233
13.2.4	閉領域作成ツール	233
13.3	経路地点操作モード	234
13.3.1	経路地点選択ツール	235
13.3.2	経路地点作成ツール	235
13.3.3	経路地点一括作成ツール	235
13.3.4	属性管理テーブル	236

13.4	エッジ操作モード	236
13.4.1	エッジ選択ツール	236
13.4.2	エッジ作成ツール	237
13.4.3	エッジ一括作成ツール	237
13.4.4	属性管理テーブル	237
13.5	画像操作モード	238
13.5.1	画像選択ツール	238
13.5.2	画像作成ツール	239
13.6	文字操作モード	239
13.6.1	文字選択ツール	239
13.6.2	文字作成ツール	240
13.7	レイヤー	240
13.8	ユーザー定義領域操作モード	241
13.8.1	領域選択ツール	242
13.8.2	矩形作成ツール	243
13.8.3	多角形作成ツール	243
13.8.4	閉領域作成ツール	244
13.8.5	属性管理テーブル	244
13.9	Undo/Redo	244
13.10	ファイルメニュー	245
13.10.1	地図初期化	245
13.10.2	地図読み込み	245
13.10.3	背景 DXF 読み込み	245
13.10.4	背景 PNG 読み込み	245
13.10.5	GeoJSON インポート	247
13.10.6	GeoJSON エクスポート	247
13.10.7	地図設定	247
13.10.8	地図保存	248
13.11	Python メニュー	248
13.11.1	Python コンソールを開く	248
第 14 章	強化学習モデル	249
14.1	強化学習モデルの編集	249
14.1.1	共通設定タブ	249
14.1.2	属性設定タブ	250
14.2	離散イベント用強化学習モデル (DiscreteEventRLModel)	251
14.3	エージェント用強化学習モデル (AgentRLModel)	253
14.4	強化学習モデルの学習モデル	256
14.4.1	行動価値関数を使用したモデル	256
14.4.2	ActorCritic モデル	260

第 15 章 3D アニメーション	269
15.1 3D アニメーション (Animation3D)	269
15.1.1 属性設定タブ	269
第 16 章 可視化ダッシュボード	271
16.1 起動と停止	271
16.1.1 準備	271
16.1.2 起動方法	272
16.1.3 停止方法	272
16.2 操作方法	272
16.2.1 構成	272
16.2.2 メニュー	272
16.2.3 グリッド	273
16.3 プロットコンポーネント	274
16.3.1 プロジェクトフォルダの選択	274
16.3.2 プロット種類の変更方法	274
16.3.3 時間スライダー	274
16.3.4 背景、経路グラフの表示設定	275
16.3.5 設定タブ	275
16.3.6 個別の描画の説明	276
16.4 テキストコンポーネント	280
16.4.1 編集/表示パネル	280
16.4.2 設定パネル	281
第 17 章 コールセンター用部品	282
17.1 資源	282
17.1.1 対応待ちリスト (CallQueue)	282
17.2 フローアイテム	282
17.2.1 オペレーター (Operator)	282
17.2.2 問い合わせ (PhoneCall)	283
17.3 部品	284
17.3.1 放棄呼 (Abandon)	284
17.3.2 時間確認 (CheckTime)	284
17.3.3 対応終了 (EndOfOperation)	284
17.3.4 受電 (GetFromCallQueue)	285
17.3.5 入電 (GetPhoneCall)	285
17.3.6 対応 (Operate)	286
17.3.7 対応待ち (PutToCallQueue)	286
17.3.8 対応開始 (StartOfOperation)	286
17.3.9 出社 (StartWorking)	287

17.3.10 退社 (StopWorking)	287
第 18 章 窓口業務用部品	288
18.1 資源	288
18.1.1 窓口 (CounterService)	288
18.2 フローアイテム	289
18.2.1 お客 (CounterServiceCustomer)	289
18.3 部品	290
18.3.1 窓口到着 (EnterCounterService)	290
18.3.2 対応終了 (LeaveCounterService)	291
18.3.3 窓口選択部品: レジ (SelectCounterServiceMultipleLines)	291
18.3.4 窓口選択部品: 切符購入窓口 (SelectCounterServiceSingleLine)	291
18.3.5 窓口利用部品: レジ (UseCounterServiceMultipleLines)	292
18.3.6 窓口利用部品: 切符購入窓口 (UseCounterServiceSingleLine)	292
第 19 章 乱数	293
19.1 乱数属性	293
19.1.1 固定	294
19.1.2 指数分布	294
19.1.3 正規分布	294
19.1.4 対数正規分布	295
19.1.5 一様分布	296
19.1.6 ベータ分布	296
19.1.7 ガンマ分布	297
19.1.8 アーラン分布	297
19.1.9 パレート分布	298
19.1.10 ワイブル分布	298
19.1.11 カイ二乗分布	299
19.1.12 F 分布	299
19.1.13 ロジスティック分布	300
19.1.14 非心カイ二乗分布	300
19.1.15 非心 F 分布	301
19.1.16 コーシー分布	302
19.1.17 レイリー分布	302
19.1.18 t 分布	303
19.1.19 三角分布	303
19.1.20 二項分布	304
19.1.21 幾何分布	304

19.1.22 超幾何分布	305
19.1.23 負の二項分布	306
19.1.24 ポアソン分布	306
19.1.25 経験分布	307
19.1.26 経験分布 (列指定、重み無し)	307
19.1.27 経験分布 (列指定、重み有り)	308
19.1.28 経験分布 (列指定、時系列)	309
19.1.29 密度分布 (列指定、重み無し)	310
19.1.30 密度分布 (列指定、重み有り)	311
19.1.31 密度分布 (列指定、時系列)	312
19.1.32 再生 (列指定)	314
19.1.33 再生 (ステップ)	314
19.1.34 再生 (変化率ステップ)	315
19.1.35 ユーザ定義ジェネレータ	316
第 20 章 キュー	318
20.1 キューの種別	318
20.1.1 FIFO キュー	318
20.1.2 優先度付きキュー	319
20.1.3 スタック	319
第 21 章 グラフ	320
21.1 定型グラフ	320
21.2 グラフ部品	323
21.3 リアルタイムグラフ部品	324
第 22 章 グラフファイル	325
22.1 グラフファイルのフォーマット	325
22.1.1 GEXF フォーマット	325
22.1.2 GraphML フォーマット	327
22.2 グラフファイルの取り込み	329
22.3 グラフファイルの書き出し	329
第 23 章 データ表示	330
23.1 サマリの表示	330
23.2 データの表示	331
23.2.1 モニター	332
23.2.2 モニターの列	332
23.2.3 時系列モニター	332
23.2.4 時系列モニターの列	332
23.3 データ表示機能	332

23.3.1 整列	332
23.3.2 検索	334
第 24 章 データ分析	336
24.1 分散分析	336
第 25 章 強化学習結果の閲覧	340
25.1 強化学習モデルビューア	340
25.1.1 設定タブ	341
25.1.2 結果タブ	342
第 26 章 アニメーション	343
26.1 パラメータ設定	343
26.2 部品パラメータ設定	345
第 27 章 VAP 連携	348
27.1 操作	348
27.1.1 配置	348
27.1.2 実行	348
27.1.3 再実行	350
27.1.4 入出力マッチング	350
27.1.5 実行結果	350
第 28 章 その他	352
28.1 Python シェルの起動	352
関連図書	353

第1章 はじめに

S⁴ Simulation System をご利用頂き、誠にありがとうございます。

S⁴ Simulation System(エスクワトロ・シミュレーション・システム)は、汎用的なシミュレーションシステムです。工場などの生産システム、サプライチェーンなどの流通システム、銀行の窓口やATM、通信システム、交通システム、コールセンターなど、確率的な振る舞いをするものを対象とするような様々な領域のシミュレーションを行う事が出来ます。こうしたら生産率が上がるのではない、コストを下げられるのではない、とお考えになっている案件を、手軽にかつ本格的にシミュレーションする事が出来ます。また、どこがボトルネックになっているか、どこで無駄なコストを発生しているのかなどの分析も同時に行う事が出来ます。S⁴ Simulation System は、簡単にシミュレーションモデルの作成と分析が出来る操作性と、他に類を見ない自由度を合わせ持つシミュレーションシステムです。

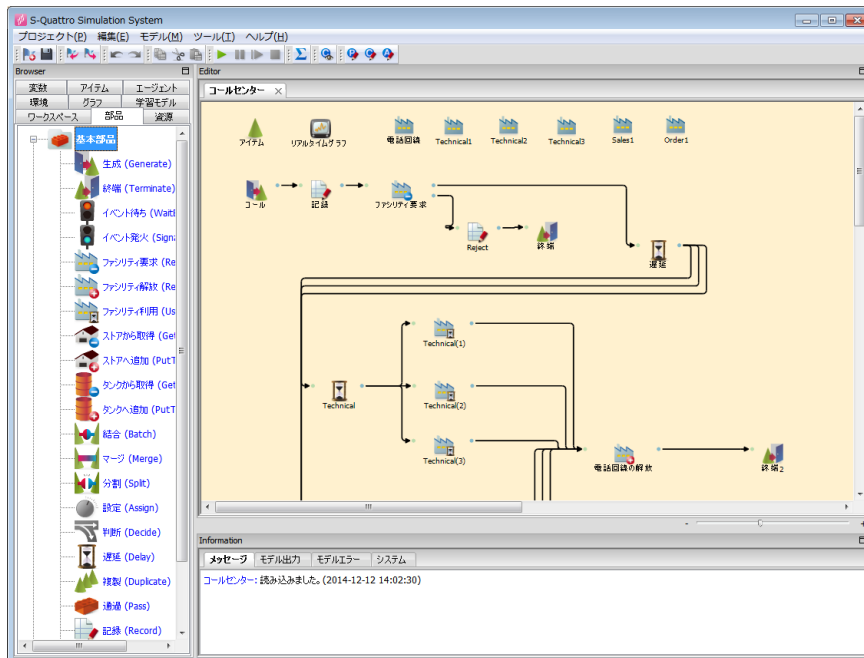


図 1.1: S⁴ Simulation System

1.1 シミュレーションについて

1.1.1 シミュレーションとは

シミュレーションとは、

現実のシステムをモデル化 (模擬) し、その振る舞いを分析・予測する問題解決手法

です。(図 1.2)

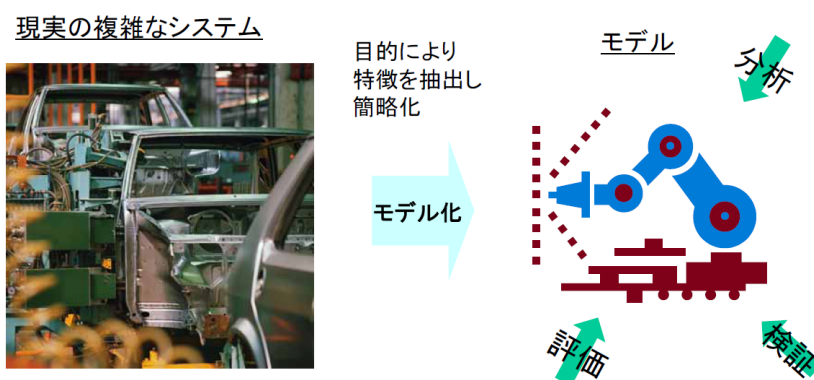


図 1.2: シミュレーションとは

1.1.2 シミュレーションのメリット

以下のような場合にシミュレーションが有効です。

- 実際のシステムで試すには時間・費用がかかる
- 実際のシステムは存在しない
- 問題が複雑で解けない
- 複数の解を比較したい

1.1.3 シミュレーションの分類

シミュレーションの種類には以下のようなものがあります。

- 連続と離散
 - － 連続型シミュレーション

- * 半導体や物理・科学シミュレーションなど
- 離散型シミュレーション
 - * 待ち行列シミュレーションなど
- 確定と確率
 - 確定的シミュレーション
 - * モデルで規定される条件に従い結果が確定的に求まるシミュレーション。連続型シミュレーションなど
 - 確率的シミュレーション
 - * システムを確率的に変動させ結果が確率的に変動するシミュレーション。待ち行列シミュレーションなどの離散イベントシミュレーション
- モンテカルロシミュレーション
 - 乱数 (モンテカルロ法) を用いてシミュレーションを行なう
 - 数値計算 (連続型・確定的シミュレーション) に用いられることもある
 - 確率 (分布) の計算 (離散型シミュレーション) に用いられることもある

S⁴ Simulation System は、確率的・離散型シミュレーションを行うツールになります。

1.1.4 離散イベントシミュレーション

離散イベントシミュレーションとは、以下のようなものを指します。

- 待ち行列タイプの現象を分析・評価するためのシミュレーション
- システムの状態変化を置こす出来事を事象 (Event) とする
- 事象の発生を乱数により表現し、待ち行列 (混雑状況など) を分析する

1.1.5 離散イベントシミュレーションの例

離散イベントシミュレーションには以下のようなものがあります。

- 工場などの生産システム (図 1.4)
- サプライチェーンなどの流通システム (図 1.5)

1.1 シミュレーションについて



図 1.3: 待ち行列のイメージ

- 銀行の窓口、ATM など (図 1.6)
- 通信システム
- 交通システム
- コールセンター
- …

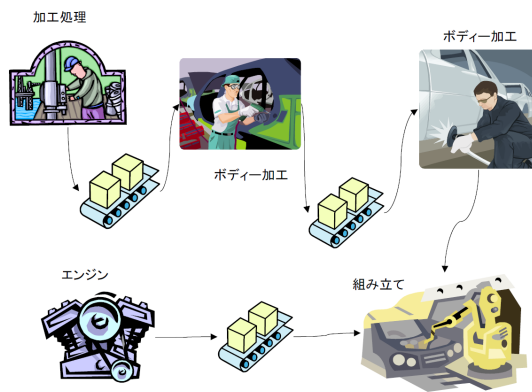


図 1.4: 工場生産の例

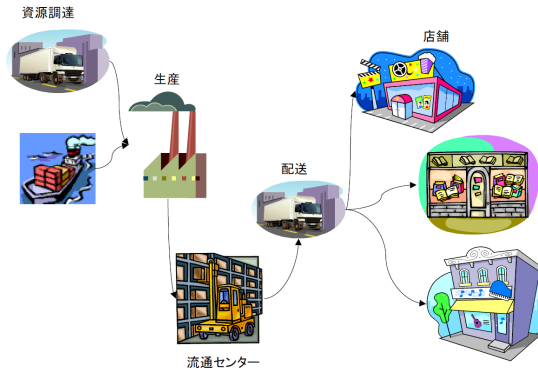


図 1.5: サプライチェーンの例

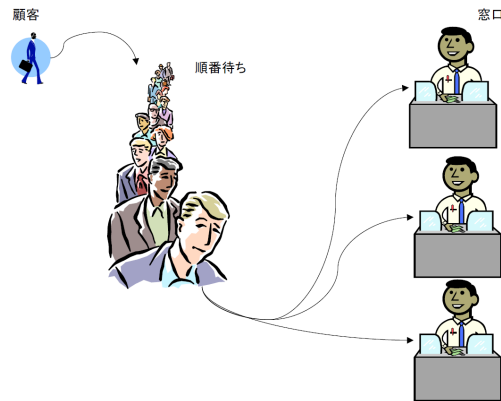


図 1.6: 銀行の窓口の例

1.2 本システムの特徴

S⁴ Simulation System は以下のような特徴を持ちます。

1.2.1 グラフィカルなモデル記述

部品を配置しリンクを貼るという手順でモデルを表現します。

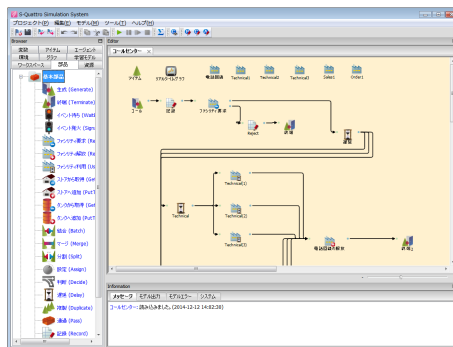


図 1.7: S⁴ Simulation System

1.2.2 部品編集によるカスタマイズ

各部品は様々なパラメータを持ち、それらを修正する事で個々の部品をモデル化します。(図 1.8, 図 1.9)

たとえば、待ち時間などには固定値を設定する事も出来ますが、様々な理論分布から選ぶ事が出来ます。また、実際に観測されたデータをインポートし、そのデータを元にサンプリングを行う事や、あるいは、そのまま再生する事で、より正確なモデル化も行えます。

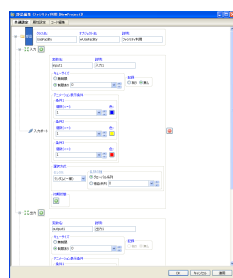


図 1.8: 部品共通設定画面

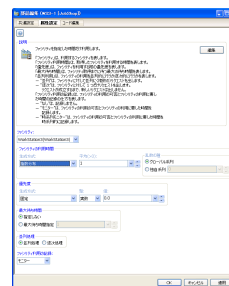


図 1.9: 部品属性設定画面

1.2.3 より高度な部品のカスタマイズ

全ての部品は、後述の psim 言語を用いて動作が記述されています。部品に用意されているパラメータでは部品の動作を正確に記述出来ない場合は、自由に書き直す事が出来ます。(図 1.10)

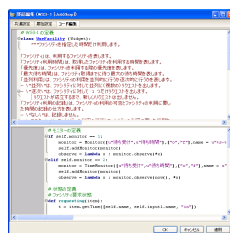


図 1.10: 部品コード編集画面

1.2.4 可視化機能

データ表示 (図 1.11)、ヒストグラム (図 1.12) や時系列プロット (図 1.13) などの様々なデータの可視化が出来ます。シミュレーション実行中にリアルタイムにグラフを表示したり、待ち行列の状態をアニメーションで表示する事も出来ます。

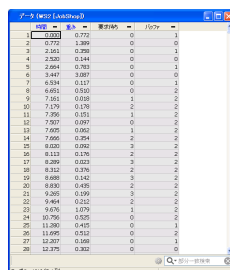


図 1.11: データ表示

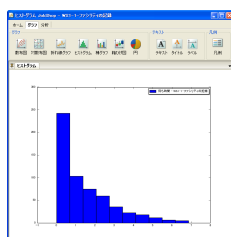


図 1.12: ヒストグラム表示

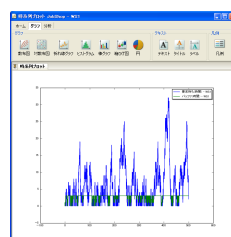


図 1.13: 時系列プロット

1.2 本システムの特徴

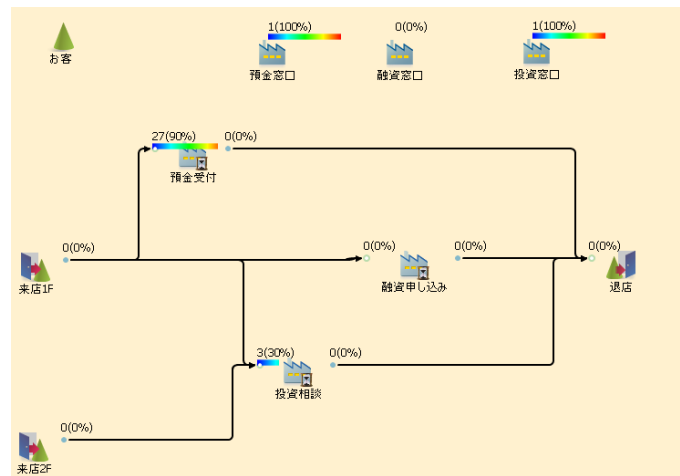


図 1.14: アニメーション

1.2.5 分析機能

各部品で観測された結果から、簡単に様々な基本統計量を得る事が出来ます (図 1.15)。独自の統計量を定義する事も出来ます (図 1.16)。また、回帰、平滑化、密度推定や分散分析などの分析を行う事が出来ます。

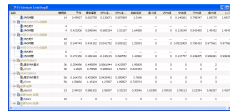


図 1.15: サマリ表示

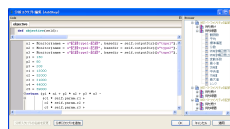


図 1.16: スクリプト編集

また、実行結果を CSV 形式で出力する事 (図 1.17) が出来ますので、外部ツールで解析する事も出来ます。Numpy, Scipy を利用した分析も可能です。

VMS との連携機能があり、VMS で作成したデータを入力してシミュレーションを実行する事や、シミュレーション結果を VMS 側に戻し分析する事も可能です。

第 1 章 はじめに

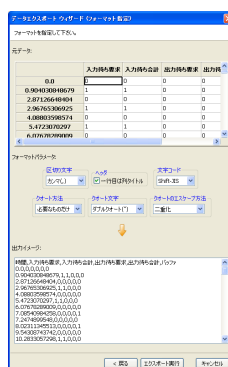


図 1.17: データエクスポート画面

1.2.6 多様な実行モード

ある目的関数の期待値を任意の線形制約の下で最大化あるいは最小化するようなパラメータセットを求める事が出来ます (図 1.18, 図 1.19)。実行中の途中結果は以下のようなグラフで表示されます (図 1.20, 図 1.21)。また、様々なパラメータで実行を行う一括実行モード、各パラメータの感度を調べる感度分析実行モード、実験計画に従って実行する実験計画実行モードも用意されています。これらの様々なパラメータで評価が必要な実行モードでは、同一 CPU 内の複数のコアを用いて、できる限り並列に実行を行います。

1.2 本システムの特徴



図 1.18: 最適化タブ



図 1.19: 条件タブ

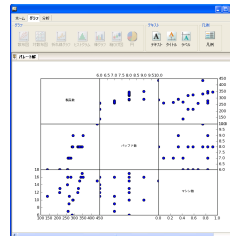
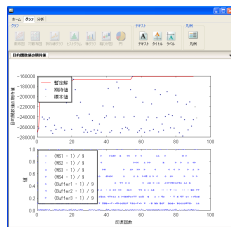


図 1.20: 最適化グラフ (単目的) 図 1.21: 最適化グラフ (多目的)

1.2.7 スケーラブルで汎用的な psim 言語

psim 言語は、Python 言語上に実装されたシミュレーション記述言語です。GUI 上で作成されたモデルは psim 言語に変換されて実行されます (図 1.22)。psim 言語を理解していなくても、シミュレーションは行えますが、psim 言語を扱う事で、より複雑なモデルも記述する事が出来ます。

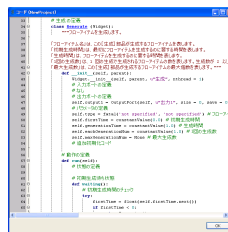


図 1.22: コード表示画面

1.2.8 ハイブリッドモデリング

離散イベントシミュレーションだけではなく、連続的に変化する状態量を扱う、感染症モデル、生態系シミュレーションなどのシステムダイナミクス

モデルも扱う事が出来ます。また、離散と連続の組合わさったハイブリッドモデルも扱う事が出来ます。

1.2.9 エージェントシミュレーション

多数の自律的に行動するエージェントの挙動から、全体の挙動をシミュレートするエージェントシミュレーションも扱う事が出来ます。各エージェントが、相互に作用し合いながら行動する事によって生じる現象のシミュレーションなど多数のエージェントからなる仮想的な世の中を構築する事が出来ます。

1.3 表記について

本ソフトウェアの正式名称は S^4 Simulation System ですが、4 乗が表記出来ない場合は **S-Quattro Simulation System** と表記する事があります。

1.4 用語

S^4 Simulation System では以下のような用語を使っています。

ワークスペース

プロジェクトファイルが格納されたディレクトリの事を指す。
ワークスペースは複数用意する事が出来、それらを切り替える事が出来る。

プロジェクト

独立したひとつのシミュレーションモデルとその入出力をまとめた区分単位。各ワークスペースに複数存在する。

モデル

シミュレーションモデルを指す。各プロジェクトにひとつだけ存在する。

入力セットフォルダ

各プロジェクトで複数の入力フォルダを持つ事が出来る。それら入力フォルダの集合を入力セットフォルダと呼ぶ。各プロジェクトにひとつだけ存在する。

出力セットフォルダ

各プロジェクトで複数の出力フォルダを持つ事が出来る。それら出力フォルダの集合を入力セットフォルダと呼ぶ。各プロジェクトにひとつだけ存在する。

入力フォルダ

外部から CSV 等をインポートし、それらを経験分布の入力としてシミュレーションを行う事が出来るが、それらのデータは入力フォルダに格納される。各入力セットフォルダに複数存在する。

出力フォルダ

モデルを実行すると、指定されている出力フォルダに結果が格納される。実行結果を別名で保存しておきたい時にも利用出来る。各出力セットフォルダに複数存在する。

モニター

実行結果あるいは、入力データが格納されている。列名を持った行列である。各入/出力フォルダに複数存在する。

時系列モニター

実行結果あるいは、入力データが格納されている。列名を持った行列であるが、モニターと違い時間列が含まれる。時間列は各行の時間を表す。基本統計量の算出方法も違う。各入/出力フォルダに複数存在する。

部品

シミュレーションモデルを構成する要素のひとつ。自律的あるいは入力を機に何らかの動作を行い、場合によっては出力を行う。モデル上に配置する事が出来る。任意個の入力ポートと出力ポートを持つ。

入力ポート

部品を構成する要素のひとつ。他の出力ポートからリンクが結ばれていればフローアイテムを入力する。キューを持つ場合と持たない場合がある。キューを持つ場合はそのキューから入力し、キューを持たない場合はセクタによってどの出力ポートから入力するかを決定する。各部品は複数の入力ポートを持つがひとつも存在しない場合もある。

出力ポート

部品を構成する要素のひとつ。他の入力ポートにリンクが結ばれていればフローアイテムを出力する。キューを持つ場合と持たない場合がある。キューを持つ場合はそのキューに出力し、キューを持たない場合はセクタによってどの入力ポートへ出力するかを決定する。各部品は複数の出力ポートを持つがひとつも存在しない場合もある。

リンク

入出力ポート間の接続を表現する。各入出力ポートに複数のリンク起点がある場合がある。

セクタ

出力ポート側にある場合は、フローアイテムを出力する際にどのリンクを選択するかを決定する方法を指す。入力ポート側にある場合は、フローアイテムを入力する際にどのリンクを選択するかを決定する方法を指す。

資源

シミュレーションモデルを構成する要素のひとつで、ファシリティ、ストア、タンク、イベントがある。シミュレーションを行う際は、その資源を利用するために待ち行列が発生するような制約要素になる。必ずキューを持つ。

ファシリティ

資源のひとつである。排他的に利用可能な複数の離散資源を持つものを表す。例えば、「銀行の窓口」など。銀行の窓口がひとつしかないとする、ある人がその窓口を占有していると、他の人はその窓口が空くのを待たなければならない。窓口を占有していた人の処理が終了すればその窓口を開放し、待ち行列に並んでいた次の人が窓口を占有する。

ストア

資源のひとつである。有限の離散量資源を貯えられるようなものを表す。例えば、「コンビニエンスストアの棚」など。コンビニエンスストアの棚には複数の商品があるが、お店の人は定期的に商品を入荷し配置するし、定期的にお客がその商品を買う。棚が一杯の時は商品を配置出来ないし、棚が空の場合は商品を買う事は出来ない。

タンク

資源のひとつである。例えば、「ガソリントank」など。ある車のガソリントankには、一定量のガソリンが入っているが、車が走ればその量は減り、ガソリンスタンドで給油すれば量が増える。容量が空の場合は車は走れず、容量が一杯の時はそれ以上給油出来ない。容量は連続量であるが、タンクへの操作は離散的にしか行えない所が、変数と異なる。

イベント

資源のひとつである。発火されるまで待ち受けする同期資源を表す。例えば、「電話がかかってくる」など。電話がかかってくるまで電話を待つが、複数の人が電話を待っている場合、並びが速いもの順に受ける場合もあり、優先度が高い順に受ける場合などがある。

フローアイテム

シミュレーション内に流れるもので、出力ポートからリンクを辿り入力ポートに入り、部品の動作の契機になる。シミュレーションモデル内には複数のフローアイテムの種類 (class) が存在し、各種類のフローアイテムの実体が実際にシミュレーション内を流れる。各フローアイテムの実体は複数の固有の属性を持つ。例えば、電話の呼出し、ベルトコンベアを流れる製品、移動する人など。

変数

連続的に変化する値を示す。参照、更新が可能な「連続変数」と、複数の連続変数の関数で表されるような「補助変数」がある。部品のひとつである連続部品は、連続変数や補助変数の値を参照し、連続的に連続変数の値を更新する。

エージェント

個々の自律的に動作する主体を表現するものである。エージェントには、固定ステップで同期的に動作する同期エージェント、個々のエージェントが非同期に動作する非同期エージェントがある。エージェントは必ずエージェント集合に属す。

エージェント集合

互いに影響を及ぼしあいながら行動する複数のエージェントの集りを表現するものである。エージェント集合は必ず環境に属す。

環境

エージェント行動の基盤となるジオメトリ情報や、全エージェントのテーブルなどを管理するものである。

同期エージェント

エージェントの一種であるが、特に全てのエージェントの状態が固定タイミングで変化するようなエージェントである。例えば、交通シミュレーションや物理シミュレーションなどで用いられる事が多い。メリットとしては、エージェント数の増加に対してスケールしやすい事がある。デメリットとしては、より精度の高いシミュレーションにするには、すべてのエージェントのステップ幅を短くする必要があり、(間隔の分散の大きい) 離散イベントの処理は、効率が悪い事などがある。

非同期エージェント

エージェントの一種であるが、特に個々のエージェントの状態が独立に離散タイミングで変化するようなエージェントである。例えば、ツイッターの伝播シミュレーション、購買行動シミュレーションなどで用いられる事が多い。メリットとしては、イベント発生間隔が疎なら、高速な離散イベントシミュレーションができる事がある。デメリットとしては、エージェント数の増加に対して、あまりスケールしない事がある。

強化学習モデル

強化学習の学習方法の設定、または学習結果のことを指す。同期エージェントの行動の決定や、フローアイテムの入出力リンクの選択の際に利用される。

第2章 起動と終了

S⁴ Simulation System の起動と終了方法についてを説明します。

2.1 起動

S⁴ Simulation System をインストールすると、Windows のスタートメニューに登録されています。

スタートメニューから、[すべてのプログラム]→[S-Quattro Simulation System]→[S-Quattro Simulation System] で起動します。

(ただし、インストール時の設定で別のメニューに登録されている場合があります。)

初回起動時は、初回起動画面 (図 2.1) が表示されます。

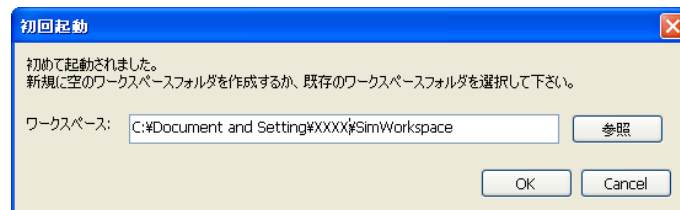


図 2.1: 初回起動

初回起動画面ではワークスペースフォルダを指定して下さい。ワークスペースフォルダとは、S⁴ Simulation System で作成したプロジェクトを保存するフォルダの事です。ワークスペースフォルダは後で変更する事も出来ます (4.3.1 章)。

なお、S⁴ Simulation System v4.4 以前で作成されたワークスペースフォルダは、v5.0 以降では使えないため、新規に指定し直す必要があります。

S⁴ Simulation System v4.4 以前に作成したプロジェクト (.sss 拡張子) の変換については、4.2.1 章を参照下さい。

2.2 終了

S⁴ Simulation System のメインウィンドウ上部のファイルメニュー (3.2.1 章) から「終了」を選ぶとで終了します。

第 2 章 起動と終了

未保存のデータやモデルが実行中の場合は、適宜確認ダイアログが表示されますので、適宜ご判断下さい。

第3章 ユーザーインターフェース

3.1 ウィンドウ構成

S⁴ Simulation System のメインウィンドウは、おおきく分けて、4 つに分かれます。(図 3.1)

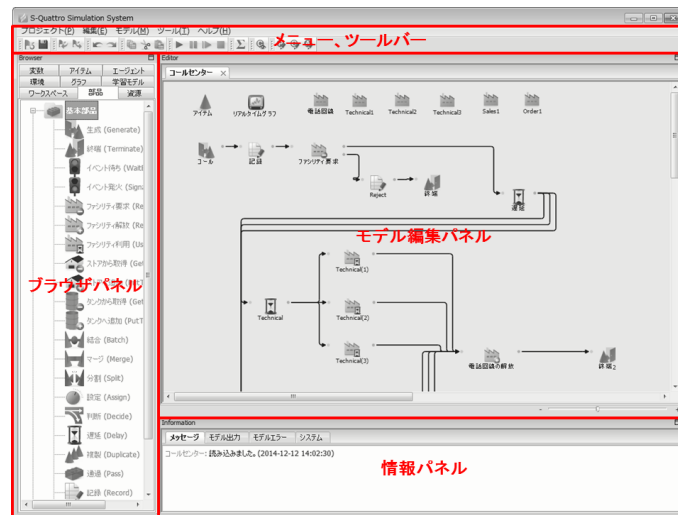


図 3.1: ウィンドウ構成

3.1.1 メニュー、ツールバー

様々な操作を行います。詳細は 3.2 章, 3.3 章 を参照して下さい。

3.1.2 ブラウザパネル

ブラウザパネルは、以下のタブから構成されます。

- ワークスペースタブ
- 部品タブ
- 資源タブ

- 変数タブ
- アイテムタブ
- グラフタブ
- エージェントタブ
- 環境タブ
- 学習モデルタブ

ワークスペースタブ

ワークスペースタブはツリー構造になっており、以下のように構成されます (図 3.2)。

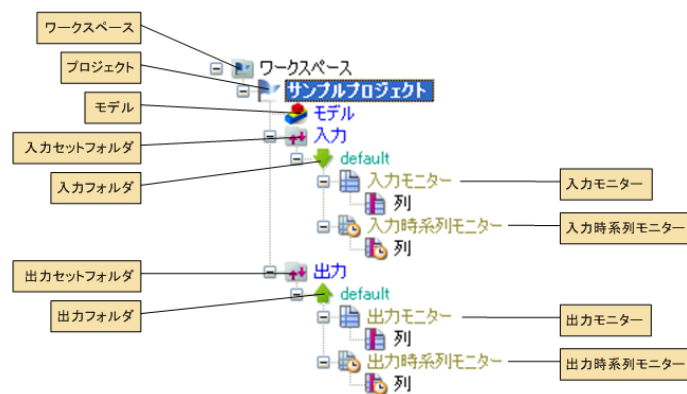


図 3.2: ワークスペースタブ

ワークスペースタブには選択されているワークスペースに含まれる全てのプロジェクトが表示されます。

各プロジェクトにはモデル、入力セットフォルダ、出力セットフォルダが含まれます。

入力セットフォルダには複数の入力フォルダ、出力セットフォルダには複数の出力フォルダが含まれます。

入出力フォルダには、複数のモニターと時系列モニターと強化学習モデルが含まれます。

モニターと時系列モニターには列が含まれます。

ワークスペースタブの各アイテムを右クリックするとメニューが表示されます。ワークスペースタブのメニューについては、3.4 章を参照して下さい。

3.1 ウィンドウ構成

部品タブ

部品タブには、部品が表示されます。各部品は、ドラッグする事でモデルに貼り付ける事が出来ます。

資源タブ

資源タブには、資源が表示されます。各資源は、ドラッグする事でモデルに貼り付ける事が出来ます。

変数タブ

変数タブには、連続変数と、補助変数が表示されます。変数は、ドラッグする事でモデルに貼り付ける事が出来ます。

アイテムタブ

アイテムタブには、フローアイテムが表示されます。フローアイテムは、ドラッグする事でモデルに貼り付ける事が出来ます。

グラフタブ

グラフタブには、グラフとリアルタイムグラフが表示されます。グラフは、ドラッグする事でモデルに貼り付ける事が出来ます。

エージェントタブ

エージェントタブには、同期エージェントと非同期エージェントが表示されます。エージェントは、ドラッグする事でモデルに貼り付ける事が出来ます。

環境タブ

環境タブには、環境が表示されます。環境は、ドラッグする事でモデルに貼り付ける事が出来ます。

学習モデルタブ

学習モデルタブには、強化学習モデルなどが表示されます。各学習モデルは、ドラッグすることでモデルに貼り付ける事が出来ます。

3.1.3 情報パネル

情報パネルは、フローティング可能で、以下の 4 つのタブから構成されます。

- メッセージタブ
- モデル出力タブ
- モデルエラータブ
- システムタブ

各タブは、情報パネル内で移動可能です。

メッセージタブ

メッセージタブにはモデルの操作が表示されます。メッセージタブで右クリックするとメニューが表示されます。メッセージタブのメニューについては、[3.6 章](#)を参照して下さい。

モデル出力タブ

モデル出力タブにはモデルの実行時の標準出力が表示されます。モデル出力タブで右クリックするとメニューが表示されます。モデル出力タブのメニューについては、[3.7 章](#)を参照して下さい。

モデルエラータブ

モデルエラータブにはモデルの実行時の標準エラー出力が表示されます。モデルエラータブで右クリックするとメニューが表示されます。モデルエラータブのメニューについては、[3.7 章](#)を参照して下さい。

システムタブ

システム上のメッセージが表示されます。システムタブで右クリックするとメニューが表示されます。システムタブのメニューについては、[3.6 章](#)を参照して下さい。

3.1.4 モデル編集パネル

活性状態のモデルが表示されます。複数のモデルを同時に開く事が出来ます。各モデルタブはモデル編集パネル内で移動可能です。

モデル編集パネルの下部にはズームコントロールがあります。ズームコントロールを右に動かすと、現在編集中のモデルが拡大されます。ズームコントロールを左に動かすと、現在編集中のモデルが縮小されます。

3.2 メニュー

S⁴ Simulation System のメインウィンドウのメニューには以下のようなものがあります。(図 3.3)



図 3.3: メニュー

3.2.1 ファイルメニュー

新規プロジェクト

新規プロジェクトを現在選択されているワークスペース作成します。詳細は 5 章を参照して下さい。

モデルを閉じる

活性状態になっているモデルを閉じます。詳細は 5 章を参照して下さい。

モデルを保存

活性状態になっているモデルを保存します。詳細は 5 章を参照して下さい。

プロジェクトを別名で保存

活性状態になっているモデルとその入力セットフォルダと出力セットフォルダを別名のプロジェクトとして保存します。詳細は 5 章を参照して下さい。

プロジェクトをインポート

エクスポートされたプロジェクトをインポートします。詳細は 4.2.1 章を参照して下さい。

プロジェクトをエクスポート

プロジェクトをエクスポートします。エクスポートしたプロジェクトはインポートする事が出来ます。詳細は 4.2.2 章を参照して下さい。

プロパティ

ワークスペースや外部エディタなど、S⁴ Simulation System 全体の振舞いを変更します。

プロパティ編集画面 (図 3.4) が表示されます。

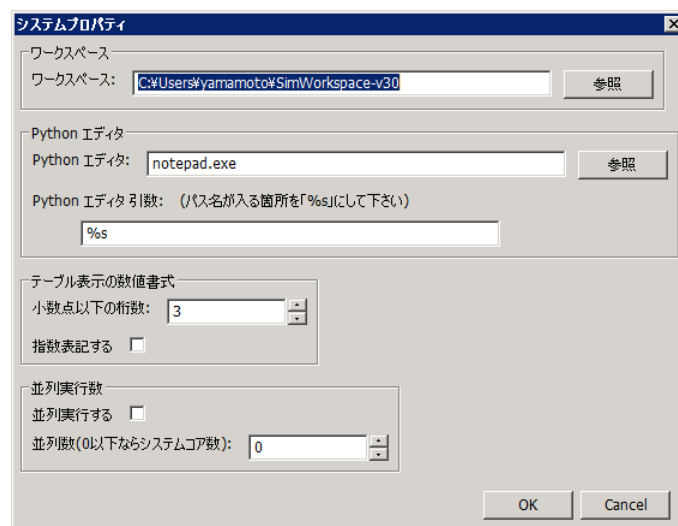


図 3.4: プロパティ

ワークスペース

ワークスペースを変更します。詳細は、[4.3.1 章](#)を参照して下さい。

Python エディタ

Python コードの参照、編集に利用する外部エディタのパスを指定します。Python エディタ引数には、外部エディタを起動する際のオプション等の引数を指定します。「%s」の部分は編集するファイル名に置換されます。([5.4.6 章](#)、[5.5 章](#)、[5.6.1 章](#)、[5.8 章](#)、[6.1.3 章](#))

テーブル表示の数値書式

テーブル表示やサマリー表示における数値の表示書式を指定します。

並列実行数

一括実行、最適化実行、感度分析実行、実験計画実行において、並列実行するかどうかを指定します。並列数を 0 にした場合、並列数はシステムの CPU のコア数になります。0 以上の数値した場合は、その数だけ並列実行します。システムコア数以上の値を設定する事はきません。

終了

S⁴ Simulation System を終了します。詳細は [2 章](#)を参照して下さい。

3.2.2 編集メニュー

元に戻す

プロジェクトを開いた後、あるいは、プロジェクトを保存後に行った操作をひとつ取り消します。詳細は [5.2 章](#)を参照して下さい。

やり直し

元に戻した操作をもう一度行います。詳細は [5.2 章](#)を参照して下さい。

切り取り

選択している部品、資源、フローアイテム、変数、エージェント、環境を切り取ります。任意のモデルに貼り付ける事が出来ます。詳細は [5章](#) を参照して下さい。

コピー

選択している部品、資源、フローアイテム、変数、エージェント、環境をコピーします。任意のモデルに貼り付ける事が出来ます。詳細は [5章](#) を参照して下さい。

貼り付け

切り取りあるいはコピーした部品、資源、フローアイテム、変数、エージェント、環境を貼り付けます。詳細は [5章](#) を参照して下さい。

3.2.3 モデルメニュー

モデルを開始する

活性状態のモデルの実行を開始します。詳細は [5.9章](#) を参照して下さい。

モデルを一時停止する

活性状態のモデルの実行を一時停止します。詳細は [5.9章](#) を参照して下さい。

モデルを再開する

一時停止している活性状態のモデルの実行を再開します。詳細は [5.9章](#) を参照して下さい。

モデルを停止する

活性状態のモデルの実行を停止します。詳細は [5.9章](#) を参照して下さい。

サマリを表示する

活性状態のモデルの出力のサマリを表示します。詳細は [23.1章](#) を参照して下さい。

コードを表示する

活性状態のモデルの自動生成されたコードを表示します。詳細は [5.8 章](#) を参照して下さい。

パラメータを編集する

活性状態のモデルのパラメータを編集します。詳細は [5.4 章](#) を参照して下さい。

カスタムコードを編集する

活性状態のモデルのカスタムコードを編集します。詳細は [5.5 章](#) を参照して下さい。

分析スクリプトを編集する

活性状態のモデルの分析スクリプトを編集します。詳細は [5.6 章](#) を参照して下さい。

モデルプロパティを編集する

活性状態のモデルのプロパティを編集します。詳細は [?? 章](#) を参照して下さい。

ナビゲーションウィンドウを表示する

ナビゲーションウィンドウを表示します。詳細は [?? 章](#) を参照して下さい。

3.2.4 ツールメニュー

Python シェル起動

Python シェルを起動します。詳細は [28.1 章](#) を参照して下さい。

3.2.5 ヘルプメニュー

S-Quattro のドキュメントを表示

S⁴ Simulation System のドキュメントを表示します。

グラフドキュメントを表示

グラフに関するドキュメントを表示します。

PDF 版のドキュメントを表示

PDF 版のドキュメントを表示します。

バージョン情報

S⁴ Simulation System のバージョン情報を表示します。

3.3 ツールバー

ツールバーにはメニューの中で良く使われるものが用意されており、グループ化されています。各グループは移動する事が出来ます。ツールバーには以下のようなものがあります。(図 3.5)



図 3.5: ツールバー

3.3.1 作成・保存

プロジェクトを新規作成

新規プロジェクトを現在選択されているワークスペース作成します。詳細は 5 章を参照して下さい。

モデルを保存する

活性状態になっているモデルを保存します。詳細は 5 章を参照して下さい。

3.3.2 インポート・エクスポート

プロジェクトをインポートする

エクスポートされたプロジェクトをインポートします。詳細は 4.2.1 章を参照して下さい。

プロジェクトをエクスポートする

プロジェクトをエクスポートします。エクスポートしたプロジェクトはインポートする事が出来ます。詳細は [4.2.2](#) 章を参照して下さい。

3.3.3 元に戻す・やり直し

元に戻す

プロジェクトを開いた後、あるいは、プロジェクトを保存後に行った操作をひとつ取り消します。詳細は [5.2](#) 章を参照して下さい。

やり直し

元に戻した操作をもう一度行います。詳細は [5.2](#) 章を参照して下さい。

3.3.4 切り取り・貼り付け

コピー

選択している部品、資源、フローアイテム、変数、エージェント、環境をコピーします。任意のモデルに貼り付ける事が出来ます。詳細は [5](#) 章を参照して下さい。

切り取り

選択している部品、資源、フローアイテム、変数、エージェント、環境を切り取ります。任意のモデルに貼り付ける事が出来ます。詳細は [5](#) 章を参照して下さい。

貼り付け

切り取りあるいはコピーした部品、資源、フローアイテム、変数、エージェント、環境を貼り付けます。詳細は [5](#) 章を参照して下さい。

3.3.5 モデル実行

モデルを開始する

活性状態のモデルの実行を開始します。詳細は [5.9](#) 章を参照して下さい。

モデルを一時停止する

活性状態のモデルの実行を一時停止します。詳細は [5.9](#) 章を参照して下さい。

モデルを再開する

一時停止している活性状態のモデルの実行を再開します。詳細は [5.9](#) 章を参照して下さい。

モデルを停止する

活性状態のモデルの実行を停止します。詳細は [5.9](#) 章を参照して下さい。

3.3.6 サマリ表示

サマリを表示

活性状態のモデルの出力のサマリを表示します。詳細は [23.1](#) 章を参照して下さい。

3.3.7 コード表示

コードを表示する

活性状態のモデルの自動生成されたコードを表示します。詳細は [5.8](#) 章を参照して下さい。

3.3.8 モデル編集

パラメータを編集する

活性状態のモデルのパラメータを編集します。詳細は [5.4](#) 章を参照して下さい。

カスタムコードを編集する

活性状態のモデルのカスタムコードを編集します。詳細は [5.5](#) 章を参照して下さい。

分析スクリプトを編集する

活性状態のモデルの分析スクリプトを編集します。詳細は 5.6 章を参照して下さい。

3.4 ワークスペースタブのメニュー

ブラウザパネルのワークスペースタブの各アイテムを右クリックするとメニューが表示されます。ワークスペースタブの構造は図 3.2 のようになっています。

3.4.1 ワークスペース

ワークスペースで右クリックすると以下のようなメニューが表示されます。

プロジェクトを新規作成する

新規プロジェクトを現在選択されているワークスペース作成します。詳細は 5 章を参照して下さい。

プロジェクトをインポートする

エクスポートされたプロジェクトをインポートします。詳細は 4.2.1 章を参照して下さい。

プロジェクトを貼りつける

コピーしたプロジェクトを貼り付ける事が出来ます。

ワークスペースを展開する

ワークスペース以下を展開します。名前の左側の「+」をクリックしても同じ動作をします。

ワークスペースを全て展開する

ワークスペース以下を再帰的に展開します。

ワークスペースを折りたたむ

ワークスペース以下を折りたたみます。名前の左側の「-」をクリックしても同じ動作をします。

ワークスペースを再読み込み

通常は必要ありませんが、ワークスペースを再読み込みします。

3.4.2 プロジェクト

プロジェクトの名前を変更する

プロジェクトの名前を変更します。

プロジェクトを削除する

プロジェクトを削除します。削除したプロジェクトは復帰出来ませんのでご注意ください。

プロジェクトをコピーする

プロジェクトをコピーします。コピーしたプロジェクトは任意のワークスペースに貼り付ける事が出来ます。

プロジェクトをエクスポートする

プロジェクトをエクスポートします。詳細は [4.2.2](#) 章を参照して下さい。

プロジェクトを展開する

プロジェクト以下を展開します。名前の左側の「+」をクリックしても同じ動作をします。

プロジェクトを折りたたむ

プロジェクト以下を折りたたみます。名前の左側の「-」をクリックしても同じ動作をします。

3.4.3 モデル

モデルで右クリックすると以下のメニューが表示されます。

モデルを開く

モデルを開きます。モデル編集パネルに表示されます。詳細は 5 章を参照して下さい。

3.4.4 入力セットフォルダ

入力セットフォルダで右クリックすると以下のメニューが表示されます。

入力フォルダを新規作成

入力フォルダを作成します。

入力セットフォルダを展開する

入力セットフォルダを展開します。名前の左側の「+」をクリックしても同じ動作をします。

入力セットフォルダを全て展開する

入力セットフォルダを再帰的に展開します。

入力セットフォルダを折りたたむ

入力セットフォルダを折りたたみます。名前の左側の「-」をクリックしても同じ動作をします。

入力セットフォルダを再読み込み

通常は必要ありませんが、入力セットフォルダを再読み込みします。

3.4.5 出力セットフォルダ

出力セットフォルダで右クリックすると以下のメニューが表示されます。

出力フォルダを新規作成

出力フォルダを作成します。

出力セットフォルダを展開する

出力セットフォルダを展開します。名前の左側の「+」をクリックしても同じ動作をします。

出力セットフォルダを全て展開する

出力セットフォルダを再帰的に展開します。

出力セットフォルダを折りたたむ

出力セットフォルダを折りたたみます。名前の左側の「-」をクリックしても同じ動作をします。

出力セットフォルダを再読み込み

通常は必要ありませんが、出力セットフォルダを再読み込みします。

3.4.6 入力フォルダ

入力フォルダで右クリックすると以下のメニューが表示されます。

データをインポートする

入力フォルダに CSV 等のテキスト形式のデータファイルをインポートします。詳細は [4.4 章](#) を参照して下さい。

入力フォルダの名前を変更する

入力フォルダの名前を変更します。

入力フォルダを削除する

入力フォルダを削除します。

入力フォルダをコピーする

入力フォルダをコピーします。コピーしたものは、入力セットフォルダあるいは出力セットフォルダに貼り付ける事が出来ます。

入力フォルダを展開する

入力フォルダを展開します。名前の左側の「+」をクリックしても同じ動作をします。

入力フォルダを全て展開する

入力フォルダを再帰的に展開します。

入力フォルダを折りたたむ

入力フォルダを折りたたみます。名前の左側の「-」をクリックしても同じ動作をします。

入力フォルダを再読み込み

通常は必要ありませんが、入力フォルダを再読み込みします。

3.4.7 出力フォルダ

出力フォルダで右クリックすると以下のメニューが表示されます。

サマリを表示

出力フォルダのサマリを表示します。詳細は [23.1](#) 章を参照して下さい。

出力フォルダの名前を変更する

出力フォルダの名前を変更します。

出力フォルダを削除する

出力フォルダを削除します。

出力フォルダをコピーする

入力フォルダをコピーします。コピーしたものは、入力セットフォルダあるいは出力セットフォルダに貼り付ける事が出来ます。

出力フォルダを展開する

出力フォルダ以下を展開します。名前の左側の「+」をクリックしても同じ動作をします。

出力フォルダを全て展開する

出力フォルダ以下を再帰的に展開します。

出力フォルダを折りたたむ

出力フォルダを折りたたみます。名前の左側の「-」をクリックしても同じ動作をします。

出力フォルダを再読み込み

通常は必要ありませんが、出力フォルダを再読み込みします。

3.4.8 モニター

モニターで右クリックすると以下のメニューが表示されます。

データを表示する

モニターの内容を表示します。詳細は [23.2 章](#)、[23.3 章](#)を参照して下さい。

プロット

モニターの各列のプロットを表示します。詳細は [21.1 章](#)を参照して下さい。

ヒストグラム

モニターの各列のヒストグラムを表示します。詳細は [21.1 章](#)を参照して下さい。

分散分析

モニターに対して分散分析を行います。詳細は [24.1](#) 章を参照して下さい。

データをエクスポートする

CSV 等のテキスト形式のデータにエクスポートします。詳細は [4.5](#) 章を参照して下さい。

データの名前を変更する

モニターの名前を変更します。

データを削除する

モニターを削除します。

データをコピーする

モニターをコピーします。コピーしたものは、入力フォルダあるいは出力フォルダに貼り付ける事が出来ます。

データを展開する

モニター以下を展開します。名前の左側の「+」をクリックしても同じ動作をします。

データを折りたたむ

モニター以下を折りたたみます。名前の左側の「-」をクリックしても同じ動作をします。

3.4.9 時系列モニター

時系列モニターで右クリックすると以下のメニューが表示されます。

データを表示する

時系列モニターの内容を表示します。時間列は「時間」列に表示されます。詳細は [23.2](#) 章、[23.3](#) 章を参照して下さい。

時系列プロット

時系列モニターの各列の時系列プロットを表示します。詳細は [21.1 章](#) を参照してください。

ヒストグラム

時系列モニターの各列のヒストグラムを表示します。時間間隔を重みとしたヒストグラムを表示します。詳細は [21.1 章](#) を参照してください。

トルネードグラフ

感度分析実行の結果の「感度分析実行結果」モニターに対してのみ表示されます。感度分析実行結果からトルネードグラフを表示します。詳細は [5.4.4 章](#) を参照して下さい。

スパイダーチャート

感度分析実行の結果の「感度分析実行結果」モニターに対してのみ表示されます。感度分析実行結果からスパイダーチャートを表示します。詳細は [5.4.4 章](#) を参照して下さい。

分散分析

時系列モニターに対して分散分析を行います。詳細は [24.1 章](#) を参照して下さい。

データをエクスポートする

CSV 等のテキスト形式のデータにエクスポートします。

データの名前を変更する

時系列モニターの名前を変更します。

データを削除する

時系列モニターを削除します。

データをコピーする

時系列モニターをコピーします。コピーしたものは、入力フォルダあるいは出力フォルダに貼り付ける事が出来ます。

データを展開する

時系列モニター以下を展開します。名前の左側の「+」をクリックしても同じ動作をします。

データを折りたたむ

時系列モニター以下を折りたたみます。名前の左側の「-」をクリックしても同じ動作をします。

3.4.10 強化学習モデル

強化学習モデルで右クリックすると以下のメニューが表示されます。

学習モデルを閲覧する

強化学習モデルが保持する学習結果を閲覧します。詳細は [25 章](#) を参照して下さい。

データの名前を変更する

強化学習モデルの名前を変更します。

データを削除する

強化学習モデルを削除します。

データをコピーする

強化学習モデルをコピーします。コピーしたものは、入力フォルダあるいは出力フォルダに貼り付ける事が出来ます。

3.4.11 列

列で右クリックすると以下のメニューが表示されます。

データを表示する

列の内容を表示します。時系列モニターの列の場合は、時間列は「時間」列に表示されます。詳細は [23.2 章](#)、[23.3 章](#) を参照して下さい。

プロット／時系列プロットを表示する

モニターの各列のプロットを表示します。時系列モニターの列の場合は、時系列プロットを表示します。詳細は [21.1 章](#) を参照して下さい。

ヒストグラムを表示する

モニターの各列のヒストグラムを表示します。時系列モニターの場合は、時系列モニターの各列のヒストグラムを表示します。時間間隔を重みとしたヒストグラムを表示します。詳細は [21.1 章](#) を参照して下さい。

3.5 モデル編集のメニュー

モデル編集画面内で、右クリックするとメニューが表示されます。

- 部品、資源、フローアイテム、変数、エージェント、環境、リンクの配置されていない場所で右クリックするとモデル編集メニューが表示されます。
- 部品、資源、フローアイテム、変数、エージェント、環境の配置されている場所で右クリックすると部品・資源・フローアイテム・変数・エージェント・環境メニューが表示されます。
- リンクのある場所で右クリックするとリンクメニューが表示されます。

3.5.1 モデル編集メニュー

モデル編集画面において、部品、資源、フローアイテム、変数、エージェント、環境、リンクの配置されていない場所で右クリックするとモデル編集メニューが表示されます。

パラメータを編集する

モデルのパラメータを編集します。詳細は [5.4 章](#) を参照して下さい。

カスタムコードを編集する

カスタムコードを編集します。詳細は [5.5 章](#) を参照して下さい。

分析スクリプトを編集する

分析スクリプトを編集します。詳細は [5.6 章](#) を参照して下さい。

コードを表示する

自動生成されたコードを表示します。詳細は [5.8 章](#) を参照して下さい。

モデルを開始する

モデルの実行を開始します。詳細は [5.9 章](#) を参照して下さい。

分析スクリプトを開始する

選択するとサブメニューに、定義されている分析スクリプト名の一覧が表示されます。指定した分析スクリプトを実行します。詳細は [5.7 章](#) を参照して下さい。

サマリを表示する

モデルの出力のサマリを表示します。詳細は [23.1 章](#) を参照して下さい。

モデルを保存する

モデルを保存します。詳細は [5 章](#) を参照して下さい。

プロジェクトをエクスポートする

このモデルを含むプロジェクトをエクスポートします。詳細は [4.2.2 章](#) を参照して下さい。

元に戻す

プロジェクトを開いた後、あるいは、プロジェクトを保存後に行った操作をひとつ取り消します。詳細は [5.2 章](#) を参照して下さい。

やり直し

元に戻した操作をもう一度行います。詳細は [5.2 章](#) を参照して下さい。

切り取り

選択している部品、資源、フローアイテム、変数、エージェント、環境を切り取ります。任意のモデルに貼り付ける事が出来ます。詳細は [5 章](#) を参照して下さい。

コピー

選択している部品、資源、フローアイテム、変数、エージェント、環境をコピーします。任意のモデルに貼り付ける事が出来ます。詳細は [5 章](#) を参照して下さい。

貼り付け

詳細は [5 章](#) を参照して下さい。

切り取りあるいはコピーした部品、資源、フローアイテム、変数、エージェント、環境を貼り付けます。

3.5.2 部品・資源・フローアイテム・変数・エージェント・環境メニュー

モデル編集画面において、部品、資源、フローアイテム、変数、エージェント、環境の配置されている場所で右クリックすると部品メニューが表示されます。

編集する

部品、資源、フローアイテム、変数、エージェント、環境を編集します。詳細は [5 章](#) を参照して下さい。

削除する

部品、資源、フローアイテム、変数、エージェント、環境を削除します。詳細は [5 章](#) を参照して下さい。

入力ポートを上 (左) に配置する

部品の入力ポートの位置を変更します。詳細は [5 章](#) を参照して下さい。

出力ポートを下 (右) に配置する

部品の出力ポートの位置を変更します。詳細は [5 章](#) を参照して下さい。

元に戻す

プロジェクトを開いた後、あるいは、プロジェクトを保存後に行った操作をひとつ取り消します。詳細は [5.2 章](#) を参照して下さい。

やり直し

元に戻した操作をもう一度行います。詳細は [5.2 章](#) を参照して下さい。

切り取り

選択している部品、資源、フローアイテム、変数、エージェント、環境を切り取ります。任意のモデルに貼り付ける事が出来ます。詳細は [5 章](#) を参照して下さい。

コピー

選択している部品、資源、フローアイテム、変数、エージェント、環境をコピーします。任意のモデルに貼り付ける事が出来ます。詳細は [5 章](#) を参照して下さい。

貼り付け

切り取りあるいはコピーした部品、資源、フローアイテム、変数、エージェント、環境を貼り付けます。詳細は [5 章](#) を参照して下さい。

3.5.3 リンクメニュー

モデル編集画面において、リンク上で右クリックするとリンクメニューが表示されます。

削除する

リンクを削除します。詳細は 5 章を参照して下さい。

リンクを隠す/表示する

リンクを隠す、あるいは、リンクを表示します。詳細は 5 章を参照して下さい。

元に戻す

プロジェクトを開いた後、あるいは、プロジェクトを保存後に行った操作をひとつ取り消します。詳細は 5.2 章を参照して下さい。

やり直し

元に戻した操作をもう一度行います。詳細は 5.2 章を参照して下さい。

切り取り

選択している部品、資源、フローアイテム、変数、エージェント、環境を切り取ります。任意のモデルに貼り付ける事が出来ます。詳細は 5 章を参照して下さい。

コピー

選択している部品、資源、フローアイテム、変数、エージェント、環境をコピーします。任意のモデルに貼り付ける事が出来ます。詳細は 5 章を参照して下さい。

貼り付け

切り取りあるいはコピーした部品、資源、フローアイテム、変数、エージェント、環境を貼り付けます。詳細は 5 章を参照して下さい。

3.6 メッセージタブ・システムタブのメニュー

情報パネルのメッセージタブとシステムタブで右クリックすると以下のメニューが表示されます。

3.6.1 クリア

タブの内容が消去されます。

3.6.2 コピー

選択されているテキストの内容がコピーされます。コピーしたテキストは他のアプリケーションに貼り付ける事が出来ます。

3.6.3 全てを選択

タブ内の全てのテキストが選択されます。

3.7 モデル出力タブ・モデルエラータブのメニュー

情報パネルのメッセージタブとシステムタブで右クリックすると以下のメニューが表示されます。

3.7.1 クリア

タブの内容が消去されます。

3.7.2 コピー

選択されているテキストの内容がコピーされます。コピーしたテキストは他のアプリケーションに貼り付ける事が出来ます。

3.7.3 全てを選択

タブ内の全てのテキストが選択されます。

3.7.4 最後のモデル出力(エラー)の結果を表示

モデル出力タブ・モデルエラータブは、最後の 1000 行だけしか表示されません。1000 行より前のテキストは随時消去されます。

このメニューを選択すると、最後のモデル出力あるいはモデルエラーの 1000 行より前を含む結果全てを表示します。ただし、色の情報は無くなってしまい、黒一色で表示されます。表示出来るのは最後に実行した結果のみです。それ以前に実行した結果は表示する事が出来ません。

3.8 Python エディタのメニュー

Python コードの編集画面 (5.4.6 章、5.5 章、5.6.1 章、5.8 章、6.1.3 章) において、右クリックすると以下のメニューが表示されます。

3.8.1 元に戻す

Python エディタ内で行った操作をひとつ取り消します。

3.8.2 やり直す

元に戻した操作をもう一度行います。

3.8.3 切り取り

選択されているテキストの内容を切り取ります。

3.8.4 コピー

選択されているテキストの内容がコピーされます。

3.8.5 貼り付け

切り取りあるいはコピーしたテキストを貼り付けます。

3.8.6 削除

選択されているテキストの内容を削除します。

3.8.7 全てを選択

全てのテキストを選択します。

3.8.8 外部エディタ

3.2.1 章で登録した外部エディタによって、Python コードを開きます。

第4章 プロジェクト

S⁴ Simulation System におけるプロジェクトとは、独立したひとつのシミュレーションモデルとその入出力をまとめた区分単位です。ここではプロジェクトの操作についてを説明します。

プロジェクトはワークスペースタブ上ではワークスペースの下の階層に表示されます。(図 4.1)

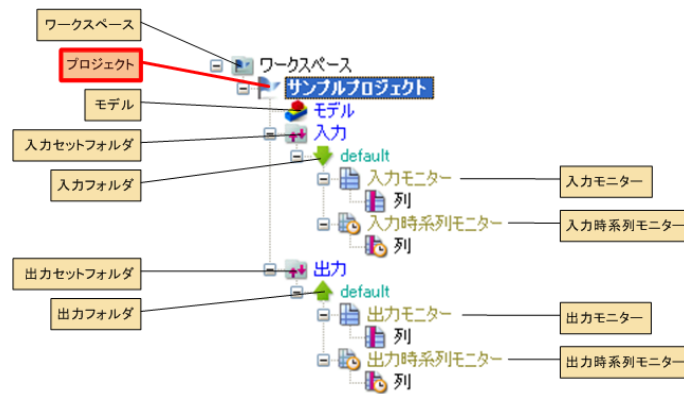


図 4.1: ワークスペースタブ上のプロジェクト

4.1 基本操作

4.1.1 プロジェクトの作成

ファイルメニュー (3.2.1 章) から「新規プロジェクト」を選択し、プロジェクトを作成します。(図 4.2)

[新規プロジェクトの作成] ダイアログ (図 4.3) が表示されますので、プロジェクト名を入力して下さい。(プロジェクト名には、Windows のファイル名に使用出来ない名前は利用出来ません。)

ブラウザパネルのワークスペースタブに作成されたプロジェクトが表示されます。(図 4.4)

モデルをダブルクリックすると、モデル編集パネルに空のモデルが表示されます。

第 4 章 プロジェクト

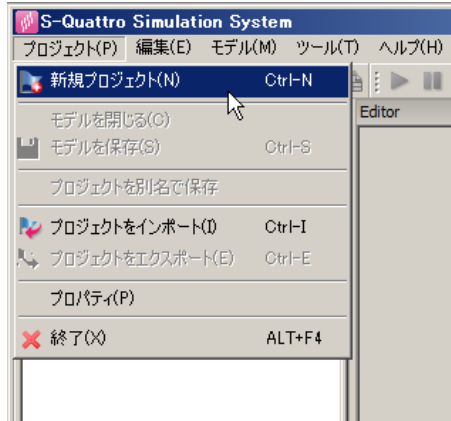


図 4.2: プロジェクトの作成

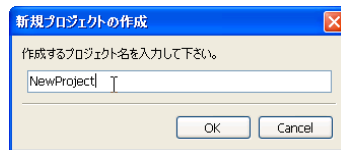


図 4.3: 新規プロジェクトダイアログ



図 4.4: 作成されたプロジェクト

4.1.2 プロジェクトの削除

ワークスペースタブ (3.1.2 章) のプロジェクトメニュー (3.4.2 章) から「プロジェクトを削除する」を選択するとプロジェクトが削除されます。削除したプロジェクトは復帰出来ませんのでご注意ください。

4.1.3 プロジェクトの名前の変更

ワークスペースタブ (3.1.2 章) のプロジェクトメニュー (3.4.2 章) から「プロジェクトの名前を変更する」を選択するとプロジェクトの名前を変更する事が出来ます。

4.2 プロジェクトのインポート・エクスポート

プロジェクトのインポート・エクスポート方法についてを説明します。

S⁴ Simulation System で作成されたプロジェクトは、外部にエクスポートする事が出来ます。エクスポートした S⁴ Simulation System のエクスポートファイルは、再度インポートする事が出来ます。

4.2.1 プロジェクトのインポート

ファイルメニュー (3.2.1 章) の「プロジェクトをインポート」を選択するとプロジェクトインポート画面が表示されます。あるいはツールバーのボタン (3.3.2 章) から実行出来ます。

プロジェクトエクスポート画面で任意のフォルダのエクスポートファイルを選択すると、現在選択されているワークスペースにインポートする事が出来ます。

S⁴ Simulation SystemV5.0 以降にエクスポートされたプロジェクトの拡張子は、.s4 となっています。

S⁴ Simulation Systemv4.4 以前にエクスポートされたプロジェクトの拡張子は、.sss となっています。v4.4 以前で作成されたプロジェクトのシミュレーションコードは、Python 2 の仕様に基づいて作成されているため .sss ファイルをインポートする場合、Python が提供する 2to3 の自動変換機能を利用して、コードが自動変換されます。しかしながら、完全に動作するコードに変換される事は保証出来ません。自動変換できなかった部分は、手でシミュレーションコードを修正する必要があります。

S⁴ Simulation Systemv4.4 のワークスペース上にあるプロジェクトをインポートしたい場合は、まず、v4.4 を起動し、当該プロジェクトをエクスポート (拡張子 .sss) します。次いで、最新の S⁴ Simulation System を起動して、旧プロジェクト (拡張子 .sss) をインポートして下さい。

4.2.2 プロジェクトのエクスポート

モデルを開いている状態で、ファイルメニュー (3.2.1 章) の「プロジェクトをエクスポート」を選択するとプロジェクトエクスポート画面が表示されます。あるいはツールバーのボタン (3.3.2 章)、モデル編集画面の部品がない場所で右クリックして表示されるメニュー (3.5.1 章) から実行出来ます。

プロジェクトエクスポート画面で任意のフォルダに任意の名前を付けて (通常は拡張子は .s4 のままにして下さい) 保存する事が出来ます。

プロジェクト単位でエクスポートしますので、モデル以外に出力セット、入力セットもまとめてエクスポートされます。

4.3 ワークスペース

4.3.1 ワークスペースの変更

ワークスペースの変更についてを説明します。

ファイルメニュー (3.2.1 章) の「プロパティ」を選択するとプロパティ画面が表示されます。

プロパティ画面には現在選択されているワークスペースフォルダが表示されています。

既存のワークスペースフォルダを指定した場合、ワークスペースフォルダが変更され、ブラウザパネルのワークスペースタブには変更されたワークスペースのプロジェクトが表示されます。

新規のワークスペースフォルダを指定した場合、ワークスペースフォルダが作成され、ブラウザパネルのワークスペースタブには空になります。

ワークスペースを変更すると、次回起動時はそのワークスペースが表示されます。

なお、S⁴ Simulation Systemv4.4 以前で作成されたワークスペースフォルダは、V5.0 以降では指定出来ません。

S⁴ Simulation Systemv4.4 以前に作成したプロジェクト (.sss 拡張子) の変換については、4.2.1 章 を参照下さい。

4.4 データのインポート

ブラウザパネルの入力フォルダを右クリックして表示されるメニュー (3.5.1 章) から「データをインポートする」を選ぶとデータインポートウィザードが表示されます。

インポートしたデータは、入力フォルダに格納され、シミュレーションのデータとして利用する事が出来ます。

以下のような手順で様々な形式のテキストデータをインポートする事が出来ます。

4.4.1 ファイル指定

最初に、インポートするテキスト形式のデータファイルを選択して下さい。

4.4.2 フォーマット指定

次にフォーマット指定画面 (図 4.5) が表示されますので、そのデータファイルのフォーマットを指定して下さい。

フォーマット指定画面は以下のような構成になっています。

ファイル内容のプレビュー

指定されたデータの先頭が表示されています。

フォーマットパラメータ

以下の様々のフォーマットパラメータが指定出来ます。

区切り文字

区切り文字を指定します。

ヘッダ

一行目が列名かデータかを指定します。

文字コード

文字コードを指定します。

クオート文字

データの中に区切り文字が含まれる場合にクオートする文字を指定します。

クオートのエスケープ方法

クオート内にクオート文字が含まれる場合のエスケープ方法を指定します。

第 4 章 プロジェクト

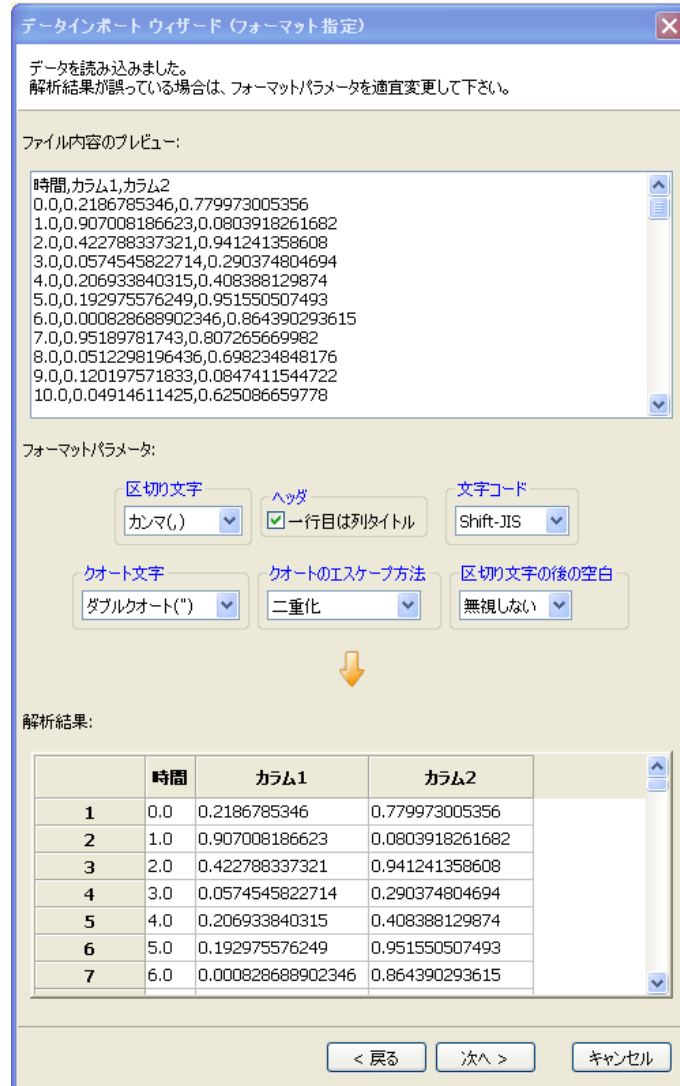


図 4.5: フォーマット指定

区切り文字の後の空白

区切り文字の後の空白の扱いを指定します。

解析結果

指定したフォーマットパラメータによりデータを解析した結果が表示されます。ここに表示されている内容が、文字化けが発生している場合や、区切り位置がおかしい、クオートの解釈の仕方が誤っているなどの時は、フォーマットパラメータを適宜修正下さい。問題がなければ、「次へ」をクリックして下さい。インポートしたいデータが間違っていた場合は「戻る」をクリックして下さい。「キャンセル」をクリックするとデータのインポートを中止します。

4.4.3 列選択

次に列選択画面 (図 4.6) が表示されますので、列を選択して下さい。
列選択画面は以下のような構成になっています。

元データ

フォーマット指定画面によって解析されたデータの先頭が表示されています。

列選択

左側には全ての列名が表示されています。

右側がインポートしたい列が表示されています。

真ん中の右矢印を押すと、左側で選択している列がインポートする列に加わります。(ただし既に追加されている列は追加出来ません。)

真ん中の左矢印を押すと、右側で選択している列がインポートする列から削除されます。

結果

列選択により選択されたデータが表示されています。問題なければ、「次へ」をクリックして下さい。問題があれば、列選択をやり直して下さい。データの解釈に誤りがある場合は「戻る」をクリックし、前の画面に戻る事が出来ます。「キャンセル」をクリックするとデータのインポートを中止します。

第 4 章 プロジェクト

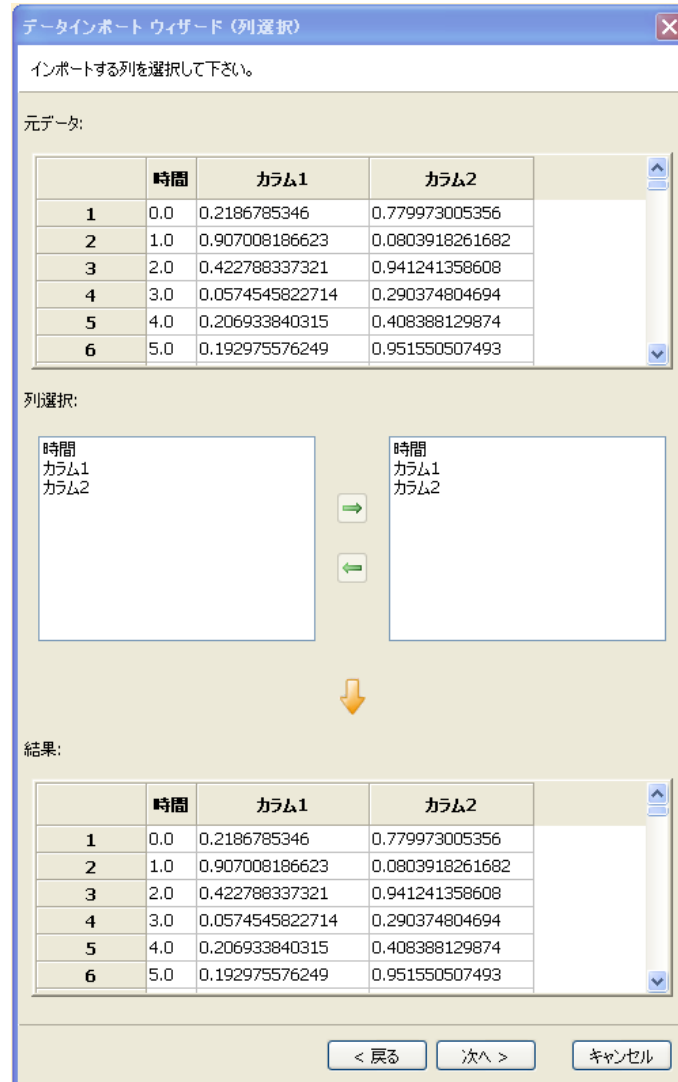


図 4.6: 列選択

4.4.4 型・名前指定

次に型・名前指定画面 (図 4.7) が表示されますので、各列の型と名前を指定して下さい。

データインポートウィザード (型・名前指定)

型と名前を指定して下さい。

元データ:

	時間	カラム1	カラム2
1	0.0	0.2186785346	0.779973005356
2	1.0	0.907008186623	0.0803918261682
3	2.0	0.422788337321	0.941241358608
4	3.0	0.0574545822714	0.290374804694
5	4.0	0.206933840315	0.408388129874
6	5.0	0.192975576249	0.951550507493
7	6.0	0.000828688902346	0.864390293615
8	7.0	0.95189781743	0.807265669982

型・名前指定:

	型	名前
時間	実数型	時間
カラム1	実数型	カラム1
カラム2	実数型	カラム2

結果:

	時間	カラム1	カラム2
1	0	0.2186785346	0.779973005356
2	1	0.907008186623	0.0803918261682
3	2	0.422788337321	0.941241358608
4	3	0.0574545822714	0.290374804694
5	4	0.206933840315	0.408388129874
6	5	0.192975576249	0.951550507493
7	6	0.000828688902346	0.864390293615
8	7	0.95189781743	0.807265669982

< 戻る 次へ > キャンセル

図 4.7: 型・名前指定

型・名前指定画面は以下のような構成になっています。

元データ

今までの指定画面によって作成されたデータの先頭が表示されています。

型・名前指定

列は型と名前で構成され、行にはデータの列名が表示されています。
データのそれぞれの列の型と名前を指定して下さい。

結果

指定した型と名前に変更されたデータが表示されています。問題なければ、「次へ」をクリックして下さい。問題があれば、型・名前指定をやり直して下さい。以前の画面での操作に誤りがある場合は「戻る」をクリックし、前の画面に戻る事が出来ます。「キャンセル」をクリックするとデータのインポートを中止します。

4.4.5 時間列指定

次に列時間列指定画面 (図 4.8) が表示されますので、時間列がある場合は指定して下さい。

時間列指定画面は以下のような構成になっています。

元データ

今までの指定画面によって作成されたデータの先頭が表示されています。

時間列指定

列は累積時間、滞在時間で構成され、行にはデータの列名が表示されています。

もし、累積時間を表している列がある場合は、その列の累積時間をチェックして下さい。

もし、滞在時間を表している列がある場合は、その列の滞在時間をチェックして下さい。

高々ひとつしかチェックする事は出来ません。

結果

指定した時間列を解釈したデータが表示されています。問題なければ、「次へ」をクリックして下さい。問題があれば、時間列指定をやり直して下さい。以前の画面での操作に誤りがある場合は「戻る」をクリックし、前の画面に戻る事が出来ます。「キャンセル」をクリックするとデータのインポートを中止します。

4.4 データのインポート

データインポート ウィザード (時間列指定)

累積時間列または滞在時間列がある場合は指定して下さい。

元データ:

	時間	カラム1	カラム2
1	0.0	0.2186785346	0.779973005356
2	1.0	0.907008186623	0.0803918261682
3	2.0	0.422788337321	0.941241358608
4	3.0	0.0574545822714	0.290374804694
5	4.0	0.206933840315	0.408388129874
6	5.0	0.192975576249	0.951550507493

時間列指定:

	累積時間	滞在時間
時間	<input type="checkbox"/>	<input type="checkbox"/>
カラム1	<input type="checkbox"/>	<input type="checkbox"/>
カラム2	<input type="checkbox"/>	<input type="checkbox"/>

↓

結果:

	時間	カラム1	カラム2
1	0	0.2186785346	0.779973005356
2	1	0.907008186623	0.0803918261682
3	2	0.422788337321	0.941241358608
4	3	0.0574545822714	0.290374804694
5	4	0.206933840315	0.408388129874
6	5	0.192975576249	0.951550507493

< 戻る 次へ > キャンセル

図 4.8: 時間列指定

4.4.6 データ名指定

最後にデータ名を指定します (図 4.9)。

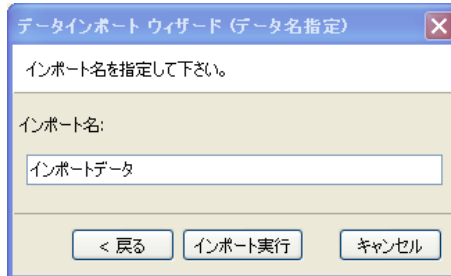


図 4.9: データ名指定

データ名を指定し「インポート実行」をクリックすると、入力フォルダに指定した名前のデータが作成されます。既にその名前のデータが存在する場合上書きを確認されます。「戻る」をクリックすると前の画面に戻ります。「キャンセル」をクリックするとデータのインポートを中止します。

4.5 データのエクスポート

ブラウザパネルの入/出力フォルダ内のデータを右クリックして表示されるメニュー (3.5.1 章) から「データをエクスポートする」を選ぶとデータエクスポートウィザードが表示されます。

エクスポートしたデータは、外部ツールで読み込む事で出来ます。

以下のような手順で様々な形式のテキストデータへエクスポートする事が出来ます。

4.5.1 ファイル指定

最初に、エクスポートするデータファイル名を選択して下さい。

4.5.2 フォーマット指定

フォーマット指定画面 (図 4.10) が表示されますので、エクスポートしたデータファイルのフォーマットを指定して下さい。

フォーマット指定画面は以下のような構成になっています。

元データ

エクスポートするデータの先頭が表示されています。

4.5 データのエクспорт



図 4.10: フォーマット指定

フォーマットパラメータ

以下の様々のフォーマットパラメータが指定出来ます。

区切り文字

区切り文字を指定します。

ヘッダ

一行目が列名かデータかを指定します。

文字コード

文字コードを指定します。

クオート方法

どのようなデータにクオートを行うかを指定します。

クオート文字

データの中に区切り文字が含まれる場合にクオートする文字を指定します。

クオートのエスケープ方法

クオート内にクオート文字が含まれる場合のエスケープ方法を指定します。

出力イメージ

指定したフォーマットパラメータによりデータを出力した場合のイメージが表示されます。ここに表示されている内容が、おかしい場合はフォーマットパラメータを適宜修正下さい。問題がなければ、「次へ」をクリックして下さい。出力するファイル名が間違っていた場合は「戻る」をクリックして下さい。「キャンセル」をクリックするとデータのエクスポートを中止します。

第5章 モデル

S⁴ Simulation System におけるモデルとは、シミュレーションモデルを指します。ひとつのプロジェクトにひとつのモデルがあります。ここではモデルの操作についてを説明します。

プロジェクトはワークスペースタブ上ではプロジェクトの下の階層に表示されます。(図 5.1)

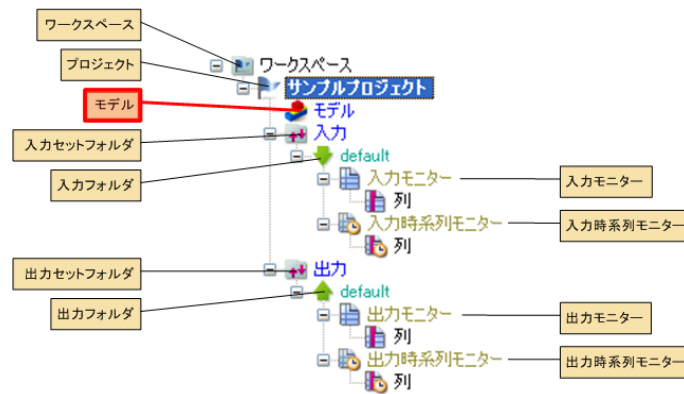


図 5.1: ワークスペースタブ上モデルプロジェクト

5.1 基本操作

5.1.1 モデルを開く

保存されているモデルあるいは新規作成したプロジェクトのモデルを開くには、ブラウザパネルのワークスペースタブのモデルをダブルクリックすると、モデル編集パネルにそのモデルが表示されます。(図 5.2)

5.1.2 モデルの保存

モデルを開いている状態で、ファイルメニュー (3.2.1 章) から、「モデルを保存」を選ぶと、モデルが保存されます。



図 5.2: モデルを開く

5.1.3 モデルを閉じる

モデル編集パネルのタブの「×」をクリックするとモデルを閉じます。(図 5.3)

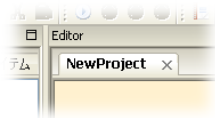


図 5.3: モデルを閉じる

5.1.4 プロジェクトを別名で保存

活性状態になっているモデルとその入力セットフォルダと出力セットフォルダを別名のプロジェクトとして保存します。入力セットフォルダと出力セットフォルダの内容はコピーされます。

5.2 元に戻す・やり直し

モデルを開いている状態で、編集メニュー (3.2.2 章) の「元に戻す」を選択するとモデルに加えた変更を取り消す事が出来ます。あるいはツールバーのボタン (3.3.3 章)、モデル編集画面で右クリックして表示されるメニュー (3.5.1 章) からも実行出来ます。

「元に戻す」は複数回の操作を取り消す事が出来ますが、モデルを開いたタイミングあるいはモデルを保存したタイミング以降の変更のみが対象となります。

「元に戻す」を行った場合、編集メニュー (3.2.2 章) の「やり直し」を選択すると「元に戻す」で取り消した操作を再度行う事が出来ます。あるいはツールバーのボタン (3.3.3 章)、モデル編集画面で右クリックして表示されるメニュー (3.5.1 章) から実行出来ます。

「やり直す」は複数回の取り消した操作を再実行する事が出来ます。

5.3 フローの編集

5.3.1 部品アイコンについて

部品アイコンは図 5.4 のように、以下で構成されます。

- アイコンイメージ
- 入力ポート
- 出力ポート
- ラベル

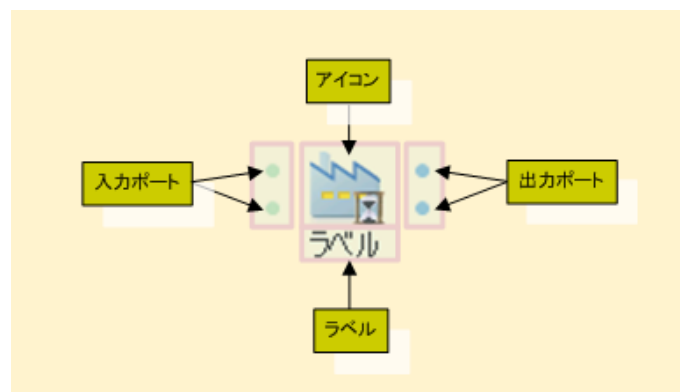


図 5.4: 部品アイコンの構成

資源アイコン、変数アイコン、フローアイテムアイコン、グラフアイコンは入/出力ポートはなく、以下のみで構成されます。

- アイコンイメージ
- ラベル

5.3.2 部品の配置

部品を配置するには、まず、モデルを開き、ブラウザパネルの部品タブを開きます。以下のような手順で配置出来ます。

1. 編集したい部品の上でマウス左ボタンを押します。
2. 左ボタンを押しながらモデル上の配置したい場所へドラッグします。
3. 配置したい場所で、左ボタンを離すと、そこに部品が配置されます。

操作イメージは、図 5.5 です。

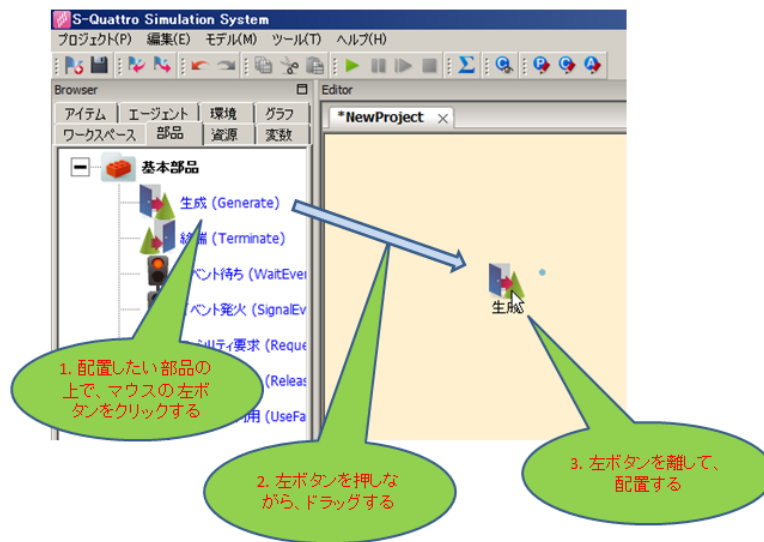


図 5.5: 部品の配置

資源を配置するには資源タブから同様の操作で配置出来ます。

フローアイテムを配置するにはアイテムタブから同様の操作で配置出来ます。

5.3.3 リンクの作成

以下のような手順で、部品間のリンクを作成出来ます。

1. リンクの起点としたい出力ポートでマウス左ボタンを押します。
2. 左ボタンを押しながらドラッグします。

3. リンクの終点としたい入力ポートで、左ボタンを離すと、リンクが作成されます。

操作イメージは、図 5.6 です。

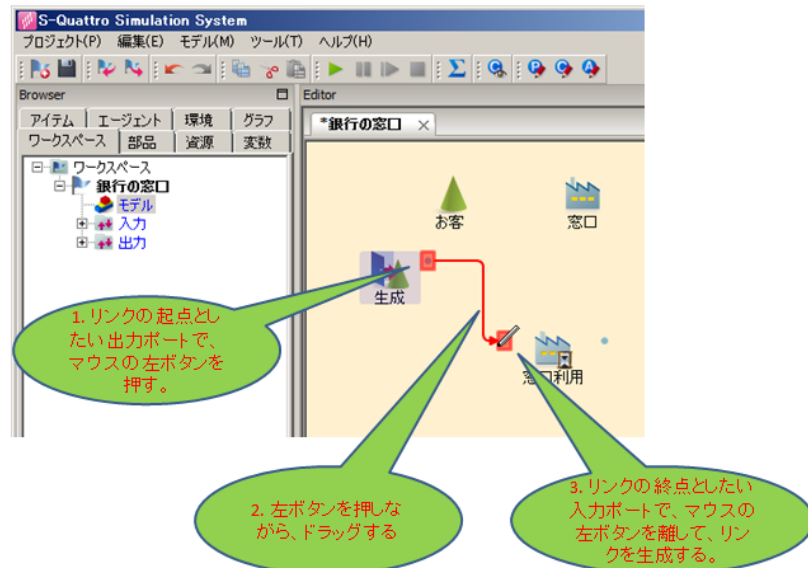


図 5.6: リンクの作成

リンクは同一モデル上の部品の出力ポートから部品の入力ポートにしか接続出来ません。リンクを作成出来る場合は、ドラッグ中にリンクが赤くなります。赤くなっていない場合、リンクは作成されません。

部品とリンクの接続方法には幾つかのパターンがあります。接続方式については、7章を参照して下さい。

5.3.4 部品の選択

部品、資源、フローアイテム、変数、エージェント、環境の配置されている場所、またはリンクのある場所を左クリックすると色が変わり選択状態になります。

Ctrl キーを押しながら部品を選択すると、部品が追加選択されます。

モデル上の部品が配置されていない場所で、マウスの左ボタンを押し、ドラッグすると長方形のガイドが表示されます。左ボタンを離すと、長方形のガイド内に含まれる全ての部品、資源、フローアイテム、変数、エージェント、環境が選択されます。Ctrl を押しながらこの操作を行うと、それらの部品が追加選択されます。

何も配置されていない場所を左クリックすると未選択状態になります。

5.3.5 部品・リンクメニュー

部品、資源、フローアイテム、変数、エージェント、環境の配置されている場所、またはリンクのある場所を右クリックするとメニューが表示されます。(図 5.7)



図 5.7: 部品・リンクメニュー

5.3.6 選択部品の移動

選択されている部品、資源、フローアイテム、変数、エージェント、環境を以下のような手順で、移動する事が出来ます。

1. 選択されている部品の上でマウス左ボタンを押します。
2. 左ボタンを押しながらモデル上の移動したい場所へドラッグします。
3. 移動したい場所で、左ボタンを離すと、そこに部品が移動します。

選択されている部品から出ているリンクも移動されます。リンクが選択されていても、そのリンクが選択されている部品につながっていない場合は移動しません。

5.3.7 選択部品の削除

部品、資源、フローアイテム、変数、エージェント、環境の配置されている場所を右クリックするとメニューが表示され、削除を選ぶと、選択状態に

ある全ての部品、資源、フローアイテム、変数、エージェント、環境が削除されます。部品の場合は、その部品から出ているリンクも削除されます。

5.3.8 リンクの削除

リンクの線上で右クリックすると、メニューが表示され、削除を選ぶと、そのリンクが削除されます。

5.3.9 リンクを隠す/表示する

リンクが表示されている時に、リンクの線上で右クリックすると、メニューが表示され、「リンクを隠す」を選ぶと、そのリンクが隠れます (図 5.8)。ただし、あくまでも表示上隠れるだけで、実際にはリンクは張られています。

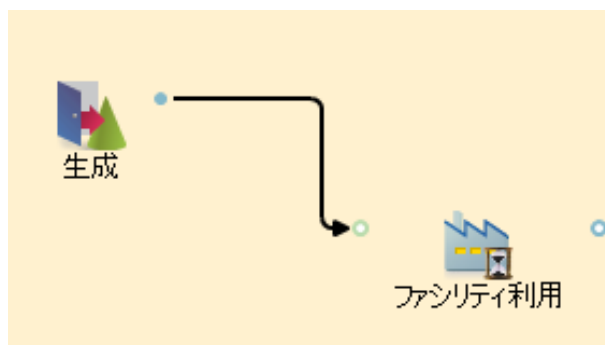


図 5.8: Link0

リンクが隠されている時に、ラベル上で右クリックすると、メニューが表示され、「リンクを表示する」を選ぶと、そのリンクが表示されます (図 5.9)。

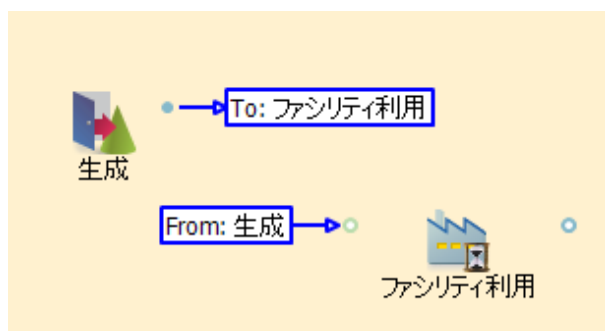


図 5.9: Link1

5.3.10 入力ポートの表示位置の変更

部品の配置されている場所を右クリックするとメニューが表示され、「入力ポートを上配置する」あるいは「入力ポートを左に配置する」を選ぶと、入力ポートの表示位置を変更できます。

5.3.11 出力ポートの表示位置の変更

部品の配置されている場所を右クリックするとメニューが表示され、「出力ポートを下に配置する」あるいは「出力ポートを右に配置する」を選ぶと、出力ポートの表示位置を変更できます。

5.3.12 選択部品の切り取り

部品、資源、フローアイテム、変数、エージェント、環境の配置されている場所を右クリックするとメニューが表示され、切り取りを選ぶと、選択状態にある全ての部品、資源、フローアイテム、変数、エージェント、環境が削除されますが、それらは後で貼り付ける事が出来ます。部品の場合は、その部品から出ているリンクも削除されますが、選択部品間のリンクは保存されており、後で貼り付けるとそれらも貼り付けられます。

5.3.13 選択部品のコピー

部品、資源、フローアイテム、変数、エージェント、環境の配置されている場所を右クリックするとメニューが表示され、コピーを選ぶと、選択状態にある全ての部品、資源、フローアイテム、変数、エージェント、環境がコピーされ、それらは後で貼り付ける事が出来ます。部品の場合は、選択部品間のリンクも保存されており、後で貼り付けるとそれらも貼り付けられます。

5.3.14 選択部品の貼り付け

部品、資源、フローアイテム、変数、エージェント、環境の配置されている場所を右クリックするとメニューが表示され、貼り付けを選ぶと、切り取りあるいはコピーされている部品、資源、フローアイテム、変数、エージェント、環境が貼り付けられます。

5.3.15 部品の名前の変更

部品、資源、フローアイテム、変数、エージェント、環境の下のラベルの場所を左ダブルクリックすると名前を変更する事が出来ます。

5.4 パラメータ編集

パラメータ編集画面では、シミュレーションモデルごとに固有のシミュレーションパラメータを編集する事が出来ます。

モデルを開いている状態で、モデルメニュー (3.2.3 章) の「パラメータを編集する」を選択するとパラメータ編集画面が表示されます。あるいはツールバーのボタン (3.3.8 章)、モデル編集画面の部品がない場所で右クリックして表示されるメニュー (3.5.1 章) から実行出来ます。

パラメータ編集画面は、

- 基本設定タブ
- パラメータタブ
- 一括実行タブ
- 感度分析タブ
- 実験計画タブ
- 最適化タブ

で構成されます。

5.4.1 基本設定タブ

基本設定タブでは、シミュレーションモデルの基本的なパラメータを設定します。(図 5.10, 図 5.11)



図 5.10: 基本設定タブ



図 5.11: 基本設定タブ (2)

実行種類

シミュレーションモデルをどのように実行するか指定します。
以下から選択できます。

通常実行

「パラメータタブ」で指定したパラメータを用いて、
1回のシミュレーションが実行されます。

一括実行

「一括実行タブ」(5.4.3章)の設定に従い、「パラ
メータタブ」で指定したパラメータを動かして、複数の
パラメータに対するシミュレーションが実行されます。

感度分析

「パラメータタブ」で指定したパラメータに対して、
基準となるパラメータと、基準からずらしたパラメ
ータに対するシミュレーションが実行されます。設定は
「感度分析タブ」で行います。

実験計画

「実験計画タブ」の設定に従い、「パラメータタブ」
で指定したパラメータを動かして、複数のパラメ
ータに対するシミュレーションが実行されます。

最適化

「最適化タブ」(5.4.6 章) の設定に従って最適化が実行されます。

シミュレーション開始時間

シミュレーション開始時間を指定します。指定無しを指定した場合は、シミュレーション開始時間を指定されません。時刻指定にした場合は、日付け、時間を指定します。

シミュレーション終了時間

シミュレーション終了時間を指定します。指定無しを選択した場合はシミュレーションを外部からは終了させません。自発的に停止しないモデルでない場合は指定してはなりません。例えば、モデル内で `finish()` を呼ぶような停止型のシミュレーションの場合は「指定なし」を選択します。時刻指定にした場合は、日付け、時間を指定します。時間指定を選択した場合は、シミュレーション時間を秒数で指定します。あるいは、時計のマークをクリックして、単位を変える事もできます。

ウォームアップ時間

シミュレーション時間が指定したウォームアップ時間になるまでモニターへの観測を行いません。定常状態を持つようなシミュレーションでは、ウォームアップ時間までのシミュレーション結果は除去する事で、定常状態に入るまでの過渡的な動作による統計の偏りを抑える事が出来ます。指定無しを選択した場合はウォームアップ時間は設定されません。時刻指定にした場合は、日付け、時間を指定します。時間指定を選択した場合は、ウォームアップ時間を秒数で指定します。あるいは、時計のマークをクリックして、単位を変える事もできます。

シミュレーション時刻フォーマット

シミュレーション時刻のフォーマットを指定します。%Y, %m, %d, %H, %M, %S, %f が指定可能です。%Y は年、%m は月、%d は日、%H は時間、%M は分、%S は秒、%f はマイクロ秒を意味します。

乱数の種 (グローバル)

グローバル系列の乱数の種を設定します。「設定しない」を選んだ場合、種はシステム時刻等から初期化されます。「設定する」を選んだ場合は指定した値が乱数の種として利用されます。

入力フォルダ

モデルを実行する際の入力フォルダを設定します。乱数属性 (19 章) において、データを引数とする時、ここで設定した入力フォルダ内のデータを取得します。

出力フォルダ

モデルを実行する際の出力フォルダを設定します。モデルを実行した結果は、ここで設定した出力フォルダに書き込まれます。古いデータは除去されます。

アニメーション

アニメーションに関するパラメータの設定をします。詳細は、(26 章) アニメーションにあります。

ODE ソルバー

微分方程式ソルバーを設定します。連続部品 (6.22 章) による連続シミュレーションは、ここで指定ソルバーが使われます。ソルバーには以下を指定する事ができます。各手法とも、時間刻み幅、相対許容誤差、絶対許容誤差を指定する事ができます。

- Dormand-Prince 5(4) 次の陽的 Runge-Kutta 法 (DOPRI54)
- Dormand-Prince 8(5,3) 次の陽的 Runge-Kutta 法 (DOP853)
- BDF 法
- Adams 法

5.4.2 パラメータタブ

パラメータタブは、シミュレーションモデルの固有パラメータを設定します。(図 5.12)

「+」を押すと末尾にパラメータが追加され、「-」を押すとパラメータが除去されます。

名前

5.4 パラメータ編集

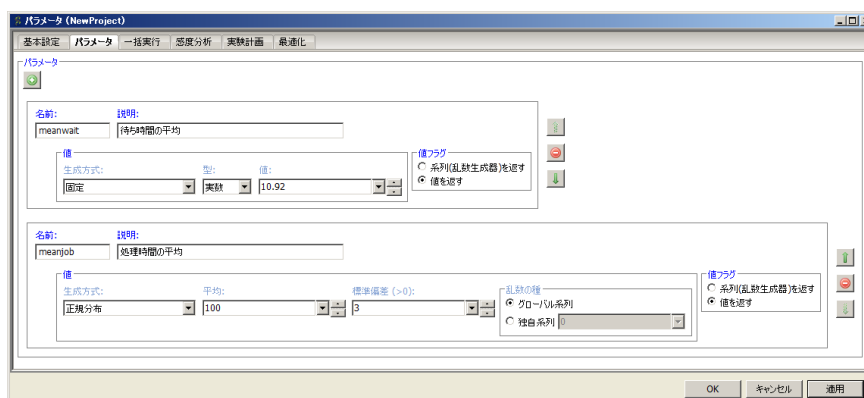


図 5.12: パラメータタブ

パラメータの名前を指定します。Python の識別子として使える名前である必要があり、1文字目がアルファベットで、2文字目以降はアルファベットか数字を指定する事が出来ます。部品の属性設定画面では、この名前でパラメータを参照する事が出来ます。

説明

パラメータの説明を指定します。モデル実行には使用されません。

値

パラメータの値を指定します。19章の乱数属性を指定する事が出来ます。

値フラグ

「系列 (乱数生成器) を返す」を指定すると、乱数発生器自体をパラメータに設定します。「値を返す」を指定すると、その乱数発生器からひとつ値を取り出し、その値をパラメータに設定します。

5.4.3 一括実行タブ

一括実行タブでは、様々なパラメータでモデルを実行する場合の設定を行います。(図 5.13)

最大反復回数

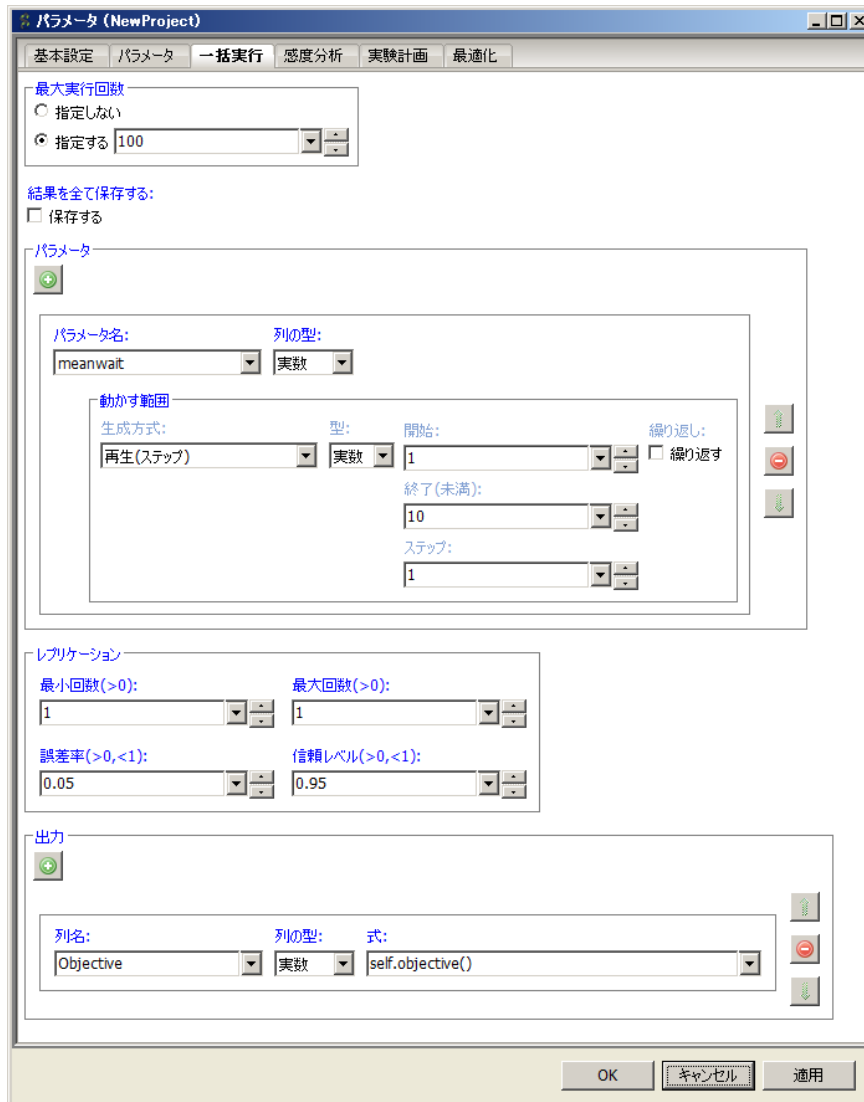


図 5.13: 一括実行タブ

実行するパラメータの個数の上限を指定します。「指定しない」を選択している場合は全てのパラメータの組合せを実行して終了します。パラメータのひとつにでも無限系列がある場合は選択してはなりません。「指定する」を選択している場合は指定された数だけのパラメータの組合せが実行されます。指定された数がパラメータの全組合せ数以上の場合は、全組合せを実行して終了します。

結果を全て保存する

個々のパラメータの組合せの実行結果を保存するか否かを指定します。保存しない場合、最後に実行した結果のみ出力フォルダ以下に保存されます。保存する場合、「出力フォルダ名-itr-NNN-RRR」という名前のサブ出力フォルダが出力フォルダ以下に作成されます。ここで、NNN は反復回数（何番目のパラメータの組合せかを表す番号）で、RRR はレプリケーション番号（何回目のレプリケーションかを表す番号）です。サブ出力フォルダには反復番号とレプリケーション番号に対応する実行結果が保存されます。

保存する・しないのどちらを選択した場合でも、「一括実行結果」と「一括実行詳細結果」というモニターが出力フォルダに作成されます。

パラメータ

一括実行でパラメータを動かす範囲を指定します。「+」を押すと末尾にパラメータが追加され、「-」を押すとパラメータが除去されます。

パラメータ名

パラメータ名を指定します。パラメータタブで定義したパラメータのみを選択する事が出来ます。

列の型

一括実行を完了すると「一括実行結果」と「一括実行詳細結果」という名前のモニターが出力フォルダに出力されます。その際のこのパラメータ列の型 (実数、整数、文字列) を指定します。

動かす範囲

動かす範囲を指定します。19 章の乱数属性を指定する事が出来ず。通常は有限系列を指定して下さい。無限系列を使う場合は高々ひとつとし、最大反復回数を設定して下さい。

レプリケーション

実行対象のパラメータの組合せを繰り返し実行する際の、繰り返し回数に関する設定を行います。

最小回数は繰り返しの最小回数、最大回数は繰り返しの最大回数を指定します。誤差率と信頼レベルは、最大回数未満で終了させる条件を定めます。具体的には、文字列型を除く全ての出力に対して、以下の条件が成立した場合は、最大回数未満で繰り返子を終了します。

誤差率を p 、信頼レベルを α とし、 n 回繰り返し実行が行われたとする。出力の標本平均を \bar{x} とし、標本標準偏差を u とし、母集団の平均値を μ とする。

この時、 μ は以下を満たさなくてはならない。

$$\bar{x} - \frac{p}{2}|\bar{x}| \leq \mu \leq \bar{x} + \frac{p}{2}|\bar{x}| \quad (5.1)$$

出力が正規分布に従うとすると、 $100\alpha\%$ 信頼区間は、以下になる。

$$\bar{x} - t_{n-1, \alpha/2} \frac{u}{\sqrt{n}} \leq \mu \leq \bar{x} + t_{n-1, \alpha/2} \frac{u}{\sqrt{n}} \quad (5.2)$$

ここで、 $t_{k, \beta}$ は自由度 k の t 分布の上側確率が β となるパーセント点をあらわす。

(5.1) 式の範囲内に (5.2) 式の信頼区間が収まるならば繰り返し回数は十分と判断する。

出力

一括実行で出力する「一括実行結果」モニターと「一括実行詳細結果」に出力する式を指定します。

「+」を押すと末尾に出力が追加され、「-」を押すと出力が除去されます。

列名

モニター上での列名を指定します。

列の型

モニター上での列の型 (実数、整数、文字列) を指定します。

式

出力する式を指定します。任意の式を書く事が出来ませんが、通常は、5.6 章の方法で分析スクリプトを定義し、その分析スクリプトを呼出すようにする方法を推奨します。

5.4.4 感度分析タブ

感度分析タブでは、1つのパラメータの組合せを基準値として、基準値と、基準値からずらしたパラメータの組合せに対してモデルを実行する場合の設定を行います。

結果を全て保存する

個々のパラメータの組合せの実行結果を保存するか否かを指定します。保存しない場合、最後に実行した結果のみ出力フォルダ以下に保存されます。保存する場合、「出力フォルダ名-itr-NNN-RRR」という名前のサブ出力フォルダが出力フォルダ以下に作成されます。ここで、NNN は反復回数 (何番目のパラメータの組合せかを表す番号) で、RRR はレプリケーション番号 (何回目のレプリケーションかを表す番号) です。サブ出力フォルダには反復番号とレプリケーション番号に対応する実行結果が保存されます。

保存する・しないのどちらを選択した場合でも、「感度分析結果」と「感度分析詳細結果」というモニターが出力フォルダに作成されます。

シミュレーション終了後グラフを表示する

感度分析実行終了後、感度分析実行の結果をあらわすグラフを自動的に表示するか否かを指定します。「表示する」にチェックを入れた場合、感度分析実行終了後に「トルネードグラフ」(図 5.15) と「スパイダーチャート」(図 5.16) の2つのグラフが表示されます。

トルネードグラフ

パラメータごとの出力の変化の幅をあらわす棒グラフです。横軸は出力の値で、パラメータごとに棒グラフが積まれます。基準値であるパラメータの組合せの

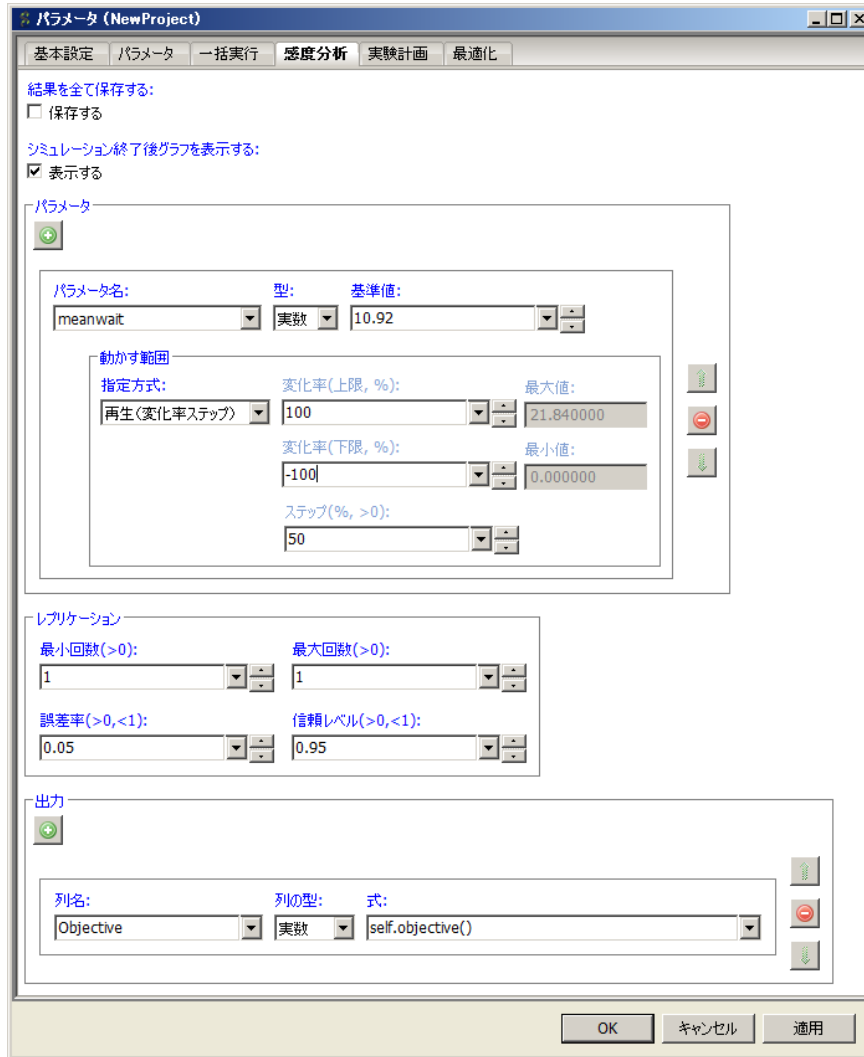


図 5.14: 感度分析タブ

5.4 パラメータ編集

出力は赤い線で表示されます。出力が複数ある場合は、出力ごとにタブに分けられてグラフが作成されます。

スパイダーチャート

パラメータの変化率と出力の関係をあらわす折れ線グラフです。横軸はパラメータの基準値からの変化率で、縦軸は出力の値です。パラメータごとに色分けされて折れ線が描かれます。出力が複数ある場合は、出力ごとにタブに分けられてグラフが作成されます。

上記の2つのグラフは、実行終了後に出力フォルダに作成される「感度分析結果」モニタの右クリックメニューからも表示できます(図 5.17)。

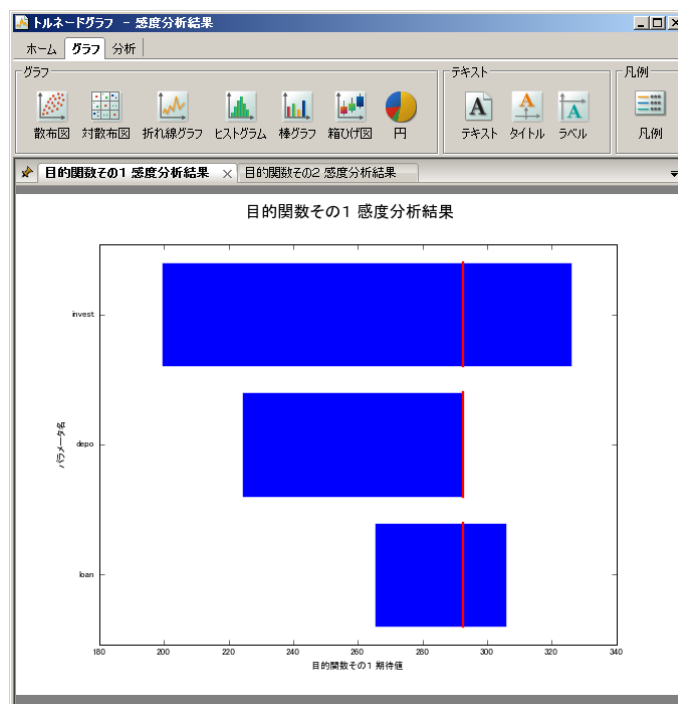


図 5.15: トルネードグラフ

パラメータ

感度分析でパラメータを動かす範囲を指定します。「+」を押すと末尾にパラメータが追加され、「-」を押すとパラメータが除去されます。

パラメータ名

第 5 章 モデル

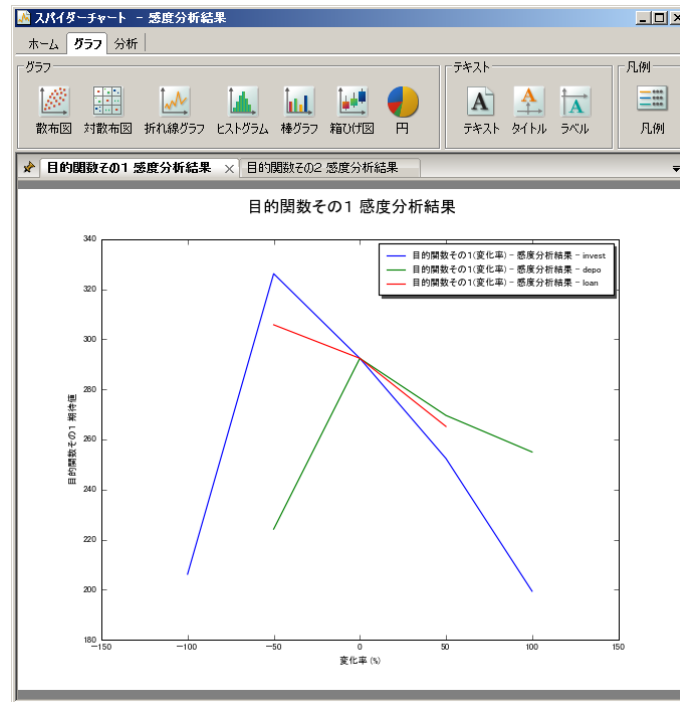


図 5.16: スパイダーチャート

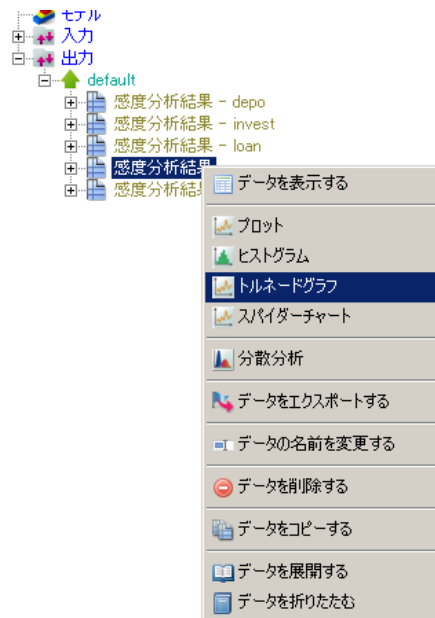


図 5.17: 「感度分析実行結果」右クリックメニュー

パラメータ名を指定します。パラメータタブ (5.4.2 章) で定義したパラメータのみを選択する事が出来ます。

型

感度分析実行を完了すると「感度分析実行結果」と「感度分析実行詳細結果」という名前のモニターが出力フォルダに出力されます。その際のこのパラメータ列の型 (実数、整数) を指定します。

動かす範囲

パラメータの動かす範囲を指定します。「再生 (変化率ステップ)」、「再生 (ステップ)」、「ユーザ定義ジェネレータ」の3つの指定方法を選べます。

再生 (変化率ステップ)

基準値からの変化率により動かし方を指定します。具体的には、基準値を b 、変化率 (上限) を α (パーセント)、変化率 (下限) を γ (パーセント)、ステップを δ (パーセント) とすると、次の値を動きます： $b(1 + \frac{-\alpha}{100}), b(1 + \frac{\delta - \alpha}{100}), b(1 + \frac{2\delta - \alpha}{100}), \dots, b(1 + \frac{\gamma}{100})$ 。

変化率 (上限)

変化率の最大値を指定します。

変化率 (下限)

変化率の最小値を指定します。

ステップ

変化率を動かす幅を指定します。

再生 (ステップ)

開始点から終了点までステップおきに動かします。

開始

開始点を指定します。

終了

終了点を指定します。(この値を含みません)

ステップ

ステップを指定します。

ユーザ定義ジェネレータ

ユーザ定義のジェネレータにより動かし方を指定します。ジェネレータの指定の仕方は、乱数のユーザ定義ジェネレータ (19.1.35 章) と同様です。

ジェネレータ

ユーザ定義のジェネレータを指定します。

最大個数

ジェネレータから生成する値の最大の個数を指定します。最大個数未満でジェネレータが値を生成を終える場合は、それ以上値を生成しません。

レプリケーション

実行対象のパラメータの組合せを繰り返し実行する際の、繰り返し回数に関する設定を行います。

最小回数は繰り返しの最小回数、最大回数は繰り返しの最大回数を指定します。誤差率と信頼レベルは、最大回数未満で終了させる条件を定めます。具体的には、文字列型を除く全ての出力に対して、以下の条件が成立した場合は、最大回数未満で繰り返しを終了します。

誤差率を p 、信頼レベルを α とし、 n 回繰り返し実行が行われたとする。出力の標本平均を \bar{x} とし、標本標準偏差を u とし、母集団の平均値を μ とする。

この時、 μ は以下を満たさなくてはならない。

$$\bar{x} - \frac{p}{2}|\bar{x}| \leq \mu \leq \bar{x} + \frac{p}{2}|\bar{x}| \quad (5.3)$$

出力が正規分布に従うとすると、 $100\alpha\%$ 信頼区間は、以下になる。

$$\bar{x} - t_{n-1, \alpha/2} \frac{u}{\sqrt{n}} \leq \mu \leq \bar{x} + t_{n-1, \alpha/2} \frac{u}{\sqrt{n}} \quad (5.4)$$

ここで、 $t_{k,\beta}$ は自由度 k の t 分布の上側確率が β となるパーセント点をあらわす。

(5.3) 式の範囲内に (5.4) 式の信頼区間が収まるならば繰り返し回数は十分と判断する。

出力

感度分析実行で出力する「感度分析実行結果」モニターと「感度分析実行詳細結果」に出力する式を指定します。

「+」を押すと末尾に出力が追加され、「-」を押すと出力が除去されます。

列名

モニター上での列名を指定します。

列の型

モニター上での列の型 (実数、整数、文字列) を指定します。

式

出力する式を指定します。任意の式を書く事が出来ませんが、通常は、5.6 章の方法で分析スクリプトを定義し、その分析スクリプトを呼出すようにする方法を推奨します。

5.4.5 実験計画タブ

実験計画タブでは、実行対象のパラメータの組合せをモニターの形で与えてモデルを実行する際の設定や、実行対象のパラメータの組合せのモニターの作成を行います。

結果を全て保存する

個々のパラメータの組合せの実行結果を保存するか否かを指定します。保存しない場合、最後に実行した結果のみ出力フォルダ以下に保存されます。保存する場合、「出力フォルダ名-ittr-NNN」という名前のサブ出力フォルダが出力フォルダ以下に作成されます。ここで、NNN は反復回数 (何番目のパラメータの組合せかを表す番号) です。サブ出力フォルダには反復番号に対応する実行結果が保存されます。

保存する・しないのどちらを選択した場合でも、「実験計画実行結果」というモニターが出力フォルダに作成されます。

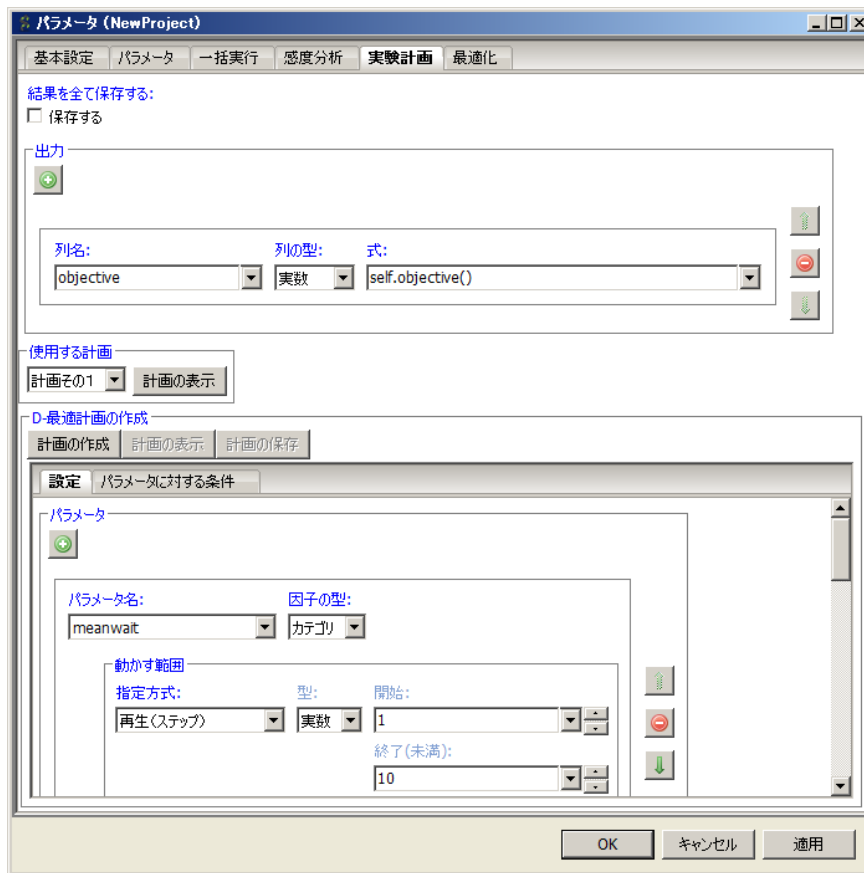


図 5.18: 実験計画タブ

出力

実験計画実行で出力する「実験計画実行結果」モニターに出力する式を指定します。

「+」を押すと末尾に出力が追加され、「-」を押すと出力が除去されます。

列名

モニター上での列名を指定します。

列の型

モニター上での列の型 (実数、整数、文字列) を指定します。

式

出力する式を指定します。任意の式を書く事が出来ますが、通常は、5.6章の方法で分析スクリプトを定義し、その分析スクリプトを呼出すようにする方法を推奨します。

使用する計画

実行対象のパラメータの組合せが記録されているモニターを選択します。入力フォルダにあるモニターから選択出来ます。モニターの各行が実行対象のパラメータの組合せとみなされ、パラメータと同名のモニターの列の値で各パラメータの値が定められます。

「計画を表示する」をクリックすると選択されたモニターが表示されます。

D-最適計画の作成

D-最適性という尺度に従って計画（実行対象のパラメータの組合せ）が作成できます。

一番上には3つのボタンが並んでいます。

計画の作成 設定に従い計画を作成します。

計画の表示 作成した計画を表示します。

計画の保存 作成した計画を入力フォルダに保存します。

ボタンの下には、以下のタブが並んでいます。

- 設定タブ
- パラメータに対する条件タブ

設定タブでは、計画の作成手法に関する設定を行います。

パラメータに対する条件タブでは、動かすパラメータの間の制約条件を設定します。

それぞれのタブの構成は以下のようになっています。

設定タブ 設定タブは図 5.19 のようになっています。

パラメータ

パラメータの取りうる値の範囲を指定します。「再生（列指定）」、「再生（ステップ）」、「再生（変化率ステップ）」、「ユーザ定義ジェネレータ」の指定方法を選べます。指定された値の範囲に含まれる値の組合せから計画が作成されます。

再生（列指定）

入力フォルダにあるモニターの列を指定し、モニターの列の値をパラメータが取りうる値とします。

値列

入力フォルダにあるモニターあるいは時系列モニターの列を選ぶ事が出来ます。図 19.10 のようにメニューから選択する事が出来ます。外部データのインポートについては、4.4 章を参照下さい。

再生（ステップ）

開始点から終了点までステップおきに動かします。

開始

開始点を指定します。

終了

終了点を指定します。(この値を含みません)

ステップ

ステップを指定します。

再生（変化率ステップ）

5.4 パラメータ編集

設定 パラメータに対する条件

パラメータ

パラメータ名: 因子の型:

動かす範囲

指定方式: 型: 開始: 終了(未満): ステップ:

パラメータ名: 因子の型:

動かす範囲

指定方式: 型: 開始: 終了(未満): ステップ:

モデル式

目的変数 =

シミュレーション実行回数:

同一のパラメータの組合せの複数回実行を許可する:
 許可する

実験作成手法設定

作成手法:

1世代あたりの個体数:

計算する世代数:

乱数の種
 設定しない
 設定する

図 5.19: 設定タブ

基準値からの変化率により動かし方を指定します。具体的には、基準値を b 、変化率 (上限) を α (パーセント)、変化率 (下限) を γ (パーセント)、ステップを δ (パーセント) とすると、次の値を動きます： $b(1 + \frac{\gamma}{100}), b(1 + \frac{\delta + \gamma}{100}), b(1 + \frac{2\delta + \gamma}{100}), \dots, b(1 + \frac{\alpha}{100})$ 。

変化率 (上限)

変化率の最大値を指定します。

変化率 (下限)

変化率の最小値を指定します。

ステップ

変化率を動かす幅を指定します。

ユーザ定義ジェネレータ

ユーザ定義のジェネレータにより動かし方を指定します。ジェネレータの指定の仕方は、乱数のユーザ定義ジェネレータ (19.1.35 章) と同様です。

ジェネレータ

ユーザ定義のジェネレータを指定します。

最大個数

ジェネレータから生成する値の最大の個数を指定します。最大個数未満でジェネレータが値を生成を終える場合は、それ以上値を生成しません。

モデル式

計画の評価尺度である D-Optimality を定める際に必要な、目的変数 (出力) に対するパラメータの回帰モデルの式を指定します。

D-Optimality は、目的変数と説明変数の値とモデル式から推定される回帰係数のばらつきの大きさを表す尺度です。D-Optimality が大きいほど (概ね) ばらつきが小さくなるため、D-Optimality を最大化するような計画が求められます。D-Optimality の具体的な定義は次のようになります。パラメータを $x = (x_1, \dots, x_n)$

とし、目的変数を y として、回帰モデルの式を $y = f(x)$ とします。この時の計画行列を X とすると、モデル式は $y = \beta X + \varepsilon$ となります。ここで β は回帰係数です。このとき、D-Optimality は $\det(X^T X)$ と定められます。

回帰モデル式はパラメータ欄で指定したパラメータ名と演算子を組合せて指定します。具体的には、モデル式には以下のものがあります。

モデル式 パラメータ名

パラメータ欄で指定したパラメータの名前は、対応するパラメータを説明変数に持つモデルを表します。

モデル式 モデル式₁+モデル式₂

モデル式₁に表れる項とモデル式₂に表れる項を説明変数に持つモデルを表します。左右に同じ項が含まれる場合はひとつの項にまとめられます。例えば $a + a$ は a と同じ意味です。

モデル式 モデル式₁-モデル式₂

モデル式₁に表れる項からモデル式₂に表れる項を除いたモデルをあらわします。例えば $(a + b) - b$ は a と同じ意味です。

モデル式 モデル式₁:モデル式₂

モデル式₁に表れる項とモデル式₂に表れる項の全ての交互作用を説明変数に持つモデルを表します。例えば、 $(a + b) : c$ は、 a と c の交互作用と b と c の交互作用を説明変数に持つモデルを表します。

モデル式 モデル式₁*モデル式₂

モデル式₁+モデル式₂+モデル式₁:モデル式₂と同じ意味です。略記用の記法です。

モデル式 モデル式**正の整数

モデル式を複数回掛けたものを表します。例えば、 $(a+b)**2$ は $(a+b)*(a+b)$ と同じ意味です。

モデル式 1

定数項を表します。ただし、明示的に 1 を書かなくても定数項はモデルに含まれます。例えば、 a と $1+a$ は同じ意味を持ちます。定数項を除きたい場合は -1 を使用します。 $a-1$ は説明変数が a のみで定数項を含まないモデルを表します。

モデル式 (モデル式₁)

モデル式₁ を表します。

モデル式 関数 (パラメータ名を含む python の式)

”関数 (パラメータ名を含む python の式)”全体を python の式として評価した結果を説明変数とするモデルをあらわします。関数には、python の組み込み関数と Numpy の関数が使用出来ます。Numpy の関数は np. 関数名の形で表します。例えば、図 5.20 の関数が使用出来ます。評価の際には、パラメータは数値 (因子の型が数値の場合) または文字列 (カテゴリの場合) のリストとなります。関数の返り値は数値または文字列のリストである必要があります。数値の場合は数値型の説明変数になり、文字列の場合はカテゴリ型の説明変数になります。

モデル式 I(パラメータ名を含む python の式)

python の式の評価結果を説明変数とするモデルを表します。式の中では、パラメータは数値（因子の型が数値の場合）または文字列（カテゴリの場合）のリストになります。式の評価結果は数値または文字列のリストである必要があります。数値の場合は数値型の説明変数になり、文字列の場合はカテゴリ型の説明変数になります。式の中では、Numpy の関数が np. 関数名の形で使用出来ます。

関数	説明
np.sin(p)	パラメータ p の各値の正弦値を返します
np.cos(p)	パラメータ p の各値の余弦値を返します
np.tan(p)	パラメータ p の各値の正接値を返します
np.exp(p)	パラメータ p の各値の指数関数値を返します
np.log(p)	パラメータ p の各値の自然対数を返します
np.sqrt(p)	パラメータ p の各値の平方根を返します
np.absolute(p)	パラメータ p の各値の絶対値を返します

図 5.20: モデル式で使用出来る関数の例

モデル式における結合優先度は、優先度の高い順に、**、:、*、+、-になります。括弧を使うことで結合優先度を制御できます。

例えば、`meanwait`、`meanjob` というパラメータがある場合は、次のようなモデル式が記述できます。

`meanwait + meanjob`

`meanwait` と `meanjob` を説明変数に持つモデルです。

`meanwait + meanjob + meanwait:meanjob`

`meanwait` と `meanjob` と 2 つの交互作用を説明変数に持つモデルです。

`np.log(meanwait) + meanjob`

meanwait の自然対数値と meanjob を説明変数に持つモデルです。このとき meanwait の因子の型は数値である必要があります。

シミュレーション実行回数

シミュレーションを行う回数を指定します。

同一のパラメータの組合せの複数回実行を許可する

許可するにチェックを入れると、各パラメータの組合せを複数回実行するような計画が作成されます。許可するにチェックを入れないと、各パラメータの組合せの実行回数は最大で 1 回になります。

実験作成手法設定

計画の作成手法の選択と、手法固有の設定を行います。作成手法は k-exchange と遺伝的アルゴリズムから選択できます。手法により設定項目が異なります。

k-exchange

k-exchange アルゴリズムに従って手法を作成します。

最大イテレーション回数

k-exchange アルゴリズム中のイテレーションの最大回数を指定します。最適と思われる計画に到達した場合¹は最大イテレーション回数未満で終了します。

k-exchange の k

k-exchange アルゴリズムのパラメータ k を指定します。1 以上シミュレーション実行回数以下の整数が指定出来ます。

乱数の種

k-exchange アルゴリズム中で使用する乱数の種を設定します。「設定しない」を選んだ場合、システム時刻等から種が定まります。「設定する」選んだ場合は、指定した値が種となります。

¹言い換えると、局所最適解である計画に収束した場合

遺伝的アルゴリズム

遺伝的アルゴリズムにより計画を作成します。

1 世代あたりの個体数

1 世代あたりの個体数を設定します。多い程良い計画が作成されますが、計算時間は長くなります。

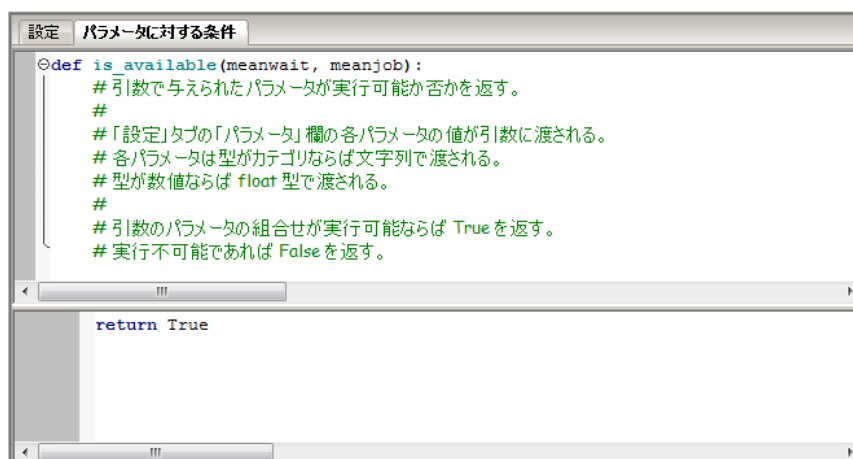
計算する世代数

計算する世代数を設定します。多い程良い計画が作成されますが、計算時間は長くなります。

乱数の種

遺伝的アルゴリズム中で使用する乱数の種を設定します。「設定しない」を選んだ場合、システム時刻等から種が定まります。「設定する」を選んだ場合は、指定した値が種となります。

パラメータに対する条件タブ パラメータに対する条件タブは図 5.21 のように 2 段構成になっています。



```

def is_available(meanwait, meanjob):
    # 引数で与えられたパラメータが実行可能か否かを返す。
    #
    # 「設定」タブの「パラメータ」欄の各パラメータの値が引数に渡される。
    # 各パラメータは型がカテゴリならば文字列で渡される。
    # 型が数値ならば float 型で渡される。
    #
    # 引数のパラメータの組合せが実行可能ならば True を返す。
    # 実行不可能であれば False を返す。

return True

```

図 5.21: パラメータに対する条件タブ

上部には、設定タブのパラメータ欄の設定内容から自動生成されたコードが表示されます。こちらは編集できません。

下部には、与えられたパラメータの組合せが実行可能かどうかを True/False で返すコードを記述します。設定タブのパラメータ欄の各パラメータの値が float 型または文字列型の値として渡されます。パラメータの型がカテゴリの場合文字列で、数値の場合 float 型です。

5.4.6 最適化タブ

最適化タブは、ある目的関数を最適にするようなシミュレーションパラメータを求めたい場合に最適化パラメータを設定します。(図 5.22)

「最適化を行う」がチェックされていない場合は何も表示されません。

「最適化を行う」がチェックされている場合は以下のタブが表示されます。

- 最適化設定タブ
- 条件タブ

最適化設定タブでは、最適化の振舞いを定める様々なパラメータを設定します。

条件タブでは、最適化変数間に制約がある場合、その制約式を設定します。それぞれのタブの構成は以下のようになっています。

最適化設定タブ

最適化変数

最適化を行う変数を定めます。パラメータ名にはシミュレーションパラメータ (5.4 章) の中から選ぶ事が出来ます。型には、整数あるいは実数を指定します。下限・上限にはその変数の取り得る範囲を指定します。

最適化

目的関数には、最適化したい目的関数を指定します。任意の式を指定する事が出来ますが、通常は、分析スクリプト 5.6 章の目的関数を指定しておきます。最大化/最小化には、目的関数を最大化したいのか最小化したいのかを指定します。多目的最適化を行う場合には複数の目的変数を指定することができます。それぞれの目的変数に対して、最大化、最小化を選択できます。

レプリケーション

5.4 パラメータ編集

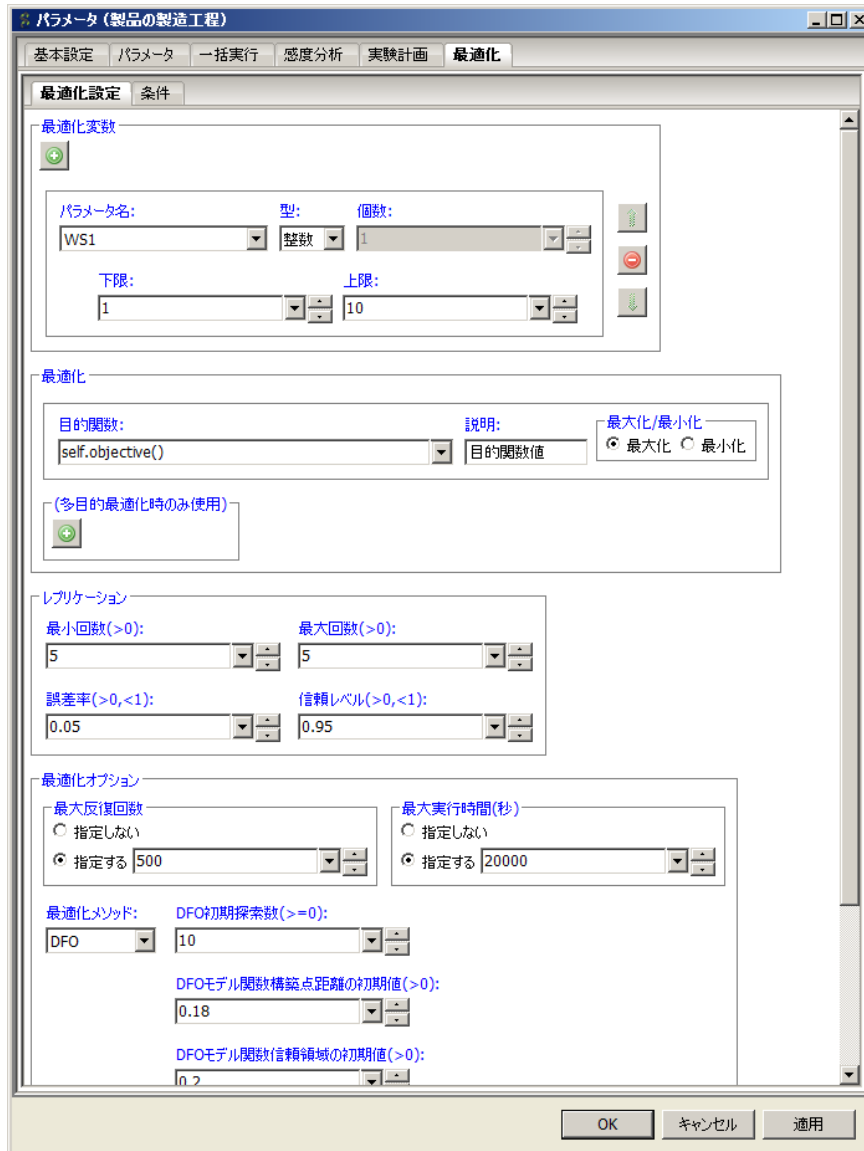


図 5.22: 最適化タブ

最小回数にはレプリケーションの最小試行回数、最大回数にはレプリケーションの最大試行回数を指定します。ただし、指定した誤差率、信頼レベルによって、以下の条件が成立した場合は、最大試行回数未満で試行が停止する場合があります。

誤差率が p で、信頼レベルを α とする。 n 個の標本を得て、標本平均を \bar{x} とし、標本標準偏差を u とし、母集団の平均値を μ とする。

この時、 μ は以下を満たさなくてはならない。

$$\bar{x} - \frac{p}{2}|\bar{x}| \leq \mu \leq \bar{x} + \frac{p}{2}|\bar{x}| \quad (5.5)$$

得られたデータを正規分布と仮定すると、 $100\alpha\%$ 信頼区間は、以下になる。

$$\bar{x} - t_{n-1, \alpha/2} \frac{u}{\sqrt{n}} \leq \mu \leq \bar{x} + t_{n-1, \alpha/2} \frac{u}{\sqrt{n}} \quad (5.6)$$

ここで、 $t_{k, \beta}$ は自由度 k の t 分布の上側確率が β となるパーセント点をあらわす。

(5.5) 式の範囲内に、(5.6) 式の信頼区間が、収まるかどうかを判定し、もし信頼区間内に収まれば、データ数は十分と判断する。収まらない場合は、追加標本を増やし、収まるまで繰り返す。

最適化オプション

最大反復数には最適化が収束しなかった場合の最大反復数を指定します。最大実行時間には最適化が収束しなかった場合の最大最適化試行時間を指定します (単位は秒)。最適化メソッドでは、DFO、PSO、ランダム探索、MOPSO の中から最適化方法を選択します。DFO と PSO とランダム探索は単目的の最適化法で、MOPSO は多目的の最適化を行います。それぞれの最適化メソッドにより指定できるパラメータが異なります。

DFO では、DFO 初期探索数、DFO モデル関数構築点距離の初期値、DFO モデル関数信頼領域の初期値、DFO 信頼領域の収束判定値を指定します。それぞれ、DFO 初期探索数には初期探索でばらまく探索数を、DFO モデル関数構築点距離の初期値には初期段階でのモデル関数構築点までの距離を、DFO モデル関数信頼領域の初期値には初期段階でのモデル関数の信頼領域の大きさを指定します。DFO 信頼領域の収束判定値には信頼領域の

大きさが指定値以下になったら、収束したと判断し最適化を停止します。

PSO、MOPSO では、Inertia weight、Accereration constant、Accereration constant、Number of Particles、Mutation toelance、Mutation Size を指定します。Inertia weight、Accereration constant は粒子の状態 (速度) の変化を指定する係数で、

$$v = wv + c_1r_1(x - x_p) + c_2r_2(x - x_g) \quad (5.7)$$

における w 、 c_1 、 c_2 を表します。Number of Particles は最適化の際に用いる粒子数を指定します。Mutation toelance は Mutation の起こしやすさを指定します。変数の範囲に対する速度の相対的な大きさを測ります。この大きさが大きいほど Mutation が起きやすくなります。Mutation Size は Mutation の大きさを指定します。

ランダム探索では、指定出来るパラメータはありません。

デバッグ出力には最適化の内部ログの有無を指定します。指定した場合、モデルエラータブに最適化のログが表示され、また、プロジェクトフォルダ (ワークスペース/projects/プロジェクト名) に残されます。

条件タブ

条件タブは、図 5.23 のように、2 段構成になっています。

上部には、最適化設定タブの最適化変数で設定した内容から自動生成されたコードが表示されています。こちらは編集する事が出来ません。

下部には、任意個の線形制約式を記述する事が出来ます。

変数宣言 変数宣言は、上部の自動生成されたコード内で記述されています。左辺値が全て変数名となっています。

式 式には以下のものがあります。

式 変数

Variable と IntegerVariable は式です。

式 数値

整数及び実数は式です。

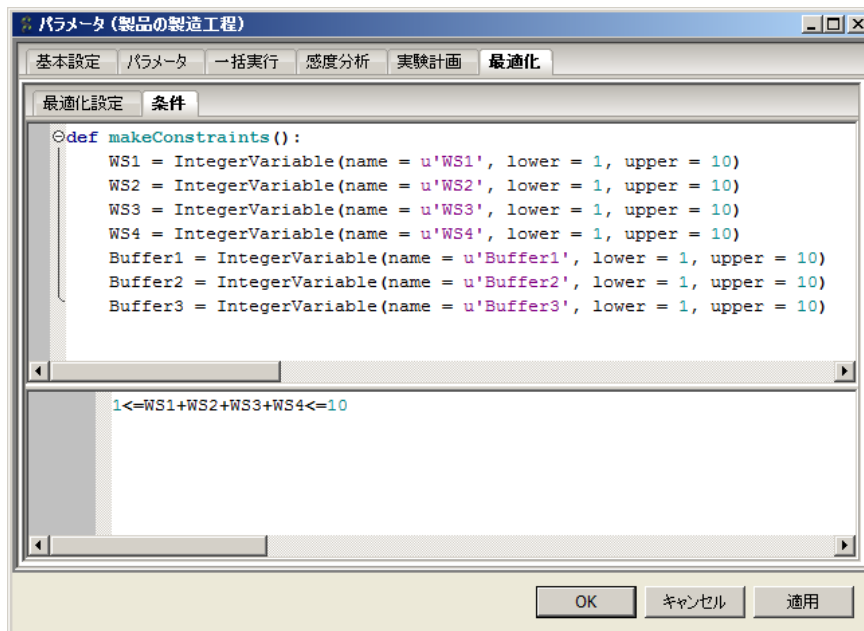


図 5.23: 条件タブ

式 $式_1 + 式_2$

式₁ と式₂ を足した式を示します。

式 $式_1 - 式_2$

式₁ から式₂ を引いた式を示します。

式 $- 式_1$

式₁ の符号を判定した式を示します。

式 $式_1 * 式_2$

式₁ と式₂ をかけた式を示します。しかし、2 次式以上の式は未対応です。つまり、少なくとも片方の式は数値でなくてはなりません。

制約式 制約式には以下のものがあります。

制約式 式₁ == 式₂

式₁ と式₂ が等しい事を示します。

制約式 式₁ <= 式₂

式₁ は式₂ 以下である事を示します。

制約式 式₁ >= 式₂

式₁ は式₂ 以上である事を示します。

制約式 (式₁)

式₁ を示します。

優先度 制約式の結合優先度は Python の演算子の優先度に従います。つまり、優先度の高い順に、<=、>=、==、+、-、*、-式、(式) になります。括弧を使う事で、結合優先度を変える事が出来ます。

例 例えば、以下のように線形制約を記述する事が出来ます。

コード例

```
def makeConstraints():
    i = IntegerVariable(name = u'i', lower = 1, upper = 6)
    j = IntegerVariable(name = u'j', lower = 3, upper = 7)
    x = Variable(name = u'x', lower = 2.0, upper = 5.0)
    y = Variable(name = u'y', lower = 3.0, upper = 4.0)
    z = Variable(name = u'z', lower = 4.0, upper = 6.0)
    # ここまでが、自動生成コード
    # 以降が、線形制約式
    i <= j
    11.0 <= x + y + z <= 13.0
    x + 2 * z <= 16.0
```

参考文献

島田直樹, 田辺隆人, 山下浩: 数式表現によらない関数の制約付き最適化, 日本オペレーションズ・リサーチ学会春季研究発表会, Vol. 2009, pp. 230-231, 2009.

5.5 カスタムコード編集

ユーザ独自のコードを編集する方法を説明します。

モデルを開いている状態で、モデルメニュー (3.2.3 章) の「カスタムコードを編集する」を選択するとカスタムコード編集画面が表示されます。あるいはツールバーのボタン、モデル編集画面の部品がない場所で右クリックして表示されるメニュー (3.5.1 章) から実行出来ます。

カスタムコード編集画面 (図 5.24) では、シミュレーションモデルごとに固有のコードを編集する事が出来ます。

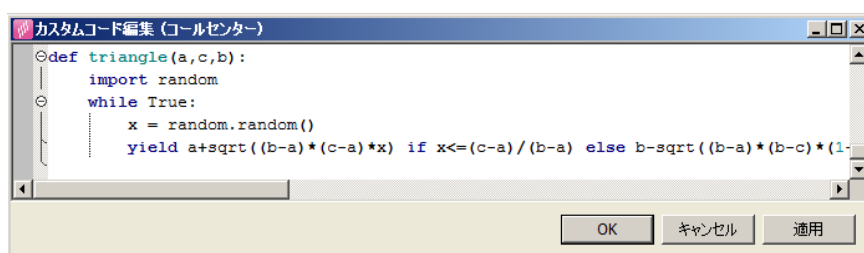


図 5.24: カスタムコード編集

カスタムコードには任意の Python コードを記述する事が出来、よく使われる用法としては、

- import 文を記述し Python の標準ライブラリあるいは、別途インストールされた拡張ライブラリをインポートしておき、部品のコード内などで、それらと呼出せるようにする。
- 独自のより複雑な理論分布や経験分布に従う乱数発生器を用意しておき、乱数属性として利用する。
- 独自の統計処理用の関数を用意しておき、分析スクリプト内で呼出せるようにする。

などがあります。

カスタムコードは自動生成コード (5.8.1 章) の先頭に挿入され、全部品から参照する事が出来ます。

5.6 分析スクリプト編集

分析スクリプト編集についてを説明します。

モデルを開いている状態で、モデルメニュー (3.2.3 章) の「分析スクリプトを編集する」を選択すると分析スクリプト編集画面が表示されます。ある

5.6 分析スクリプト編集

いはツールバーのボタン、モデル編集画面の部品がない場所で右クリックして表示されるメニュー (3.5.1 章) から実行出来ます。

分析スクリプト編集画面 (図 5.25) では、シミュレーションモデルごとに固有の分析スクリプトを編集する事が出来ます。

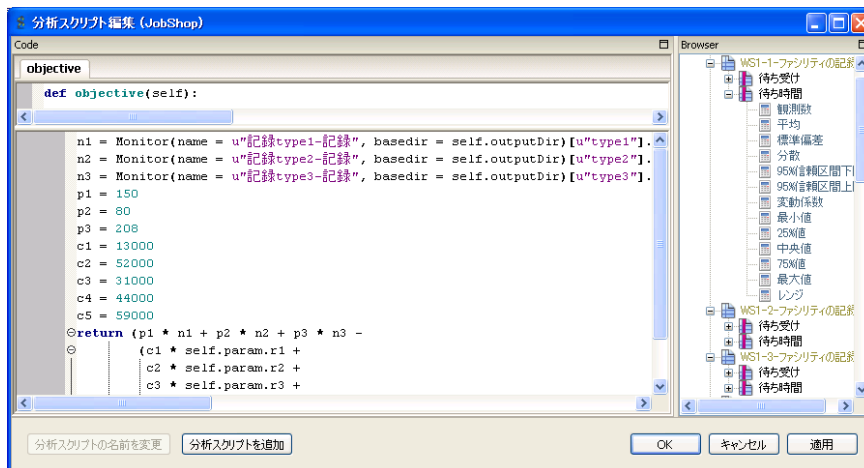


図 5.25: スクリプト編集画面

作成した分析スクリプトの実行については、5.7 章を参照下さい。

分析スクリプト画面は、左右に 2 つの領域に分かれます。

左 スクリプト定義エディタ。自由に編集する事が出来ます。

右 データブラウザ。

5.6.1 スクリプト定義エディタ

スクリプト定義エディタは複数のスクリプトを編集出来ます。各スクリプトには名前がありその名前がタブ名になっています。タブをクリックすると表示されるスクリプトが変わります。

「objective」という名前のスクリプトはモデルに必ずひとつ存在します。モデル実行時にこのスクリプトが実行され、情報パネルのモデル出力タブにその結果が表示されます。新規に作成したプロジェクトでは、「return 0」となっているので実行すると常に 0 が表示されます。

スクリプト定義エディタの上部には、スクリプトの定義ヘッダが表示され、こちらは編集する事が出来ません。

スクリプト定義エディタの下部には、スクリプトの定義本体があり、こちらは自由に編集する事が出来ます。任意の Python コードを記述する事が出来ますが、必ず最後には return 文を書く必要があります。(Python の仕様で、return 文がない場合は常に None が返されます。)

分析スクリプトの名前を変更

表示されている分析スクリプトの名前を変更します。Python の識別子として使える名前である必要があり、1 文字目がアルファベットで、2 文字目以降はアルファベットか数字を指定する事が出来ます。その他、予約語で仕様出来ない名前もあります。エラーが表示された場合は別の名前を指定して下さい。また、objective は予約語であるため名前を変更する事が出来ません。

分析スクリプトを追加

分析スクリプトを追加する事が出来ます。Python の識別子として使える名前である必要があり、1 文字目がアルファベットで、2 文字目以降はアルファベットか数字を指定する事が出来ます。その他、予約語で仕様出来ない名前もあります。エラーが表示された場合は別の名前を指定して下さい。

スクリプト定義には任意の Python 式を書く事が出来ますが、基本的な統計量はデータブラウザから指定する事が出来ます。

5.6.2 データブラウザ

データブラウザには、モデルの出力フォルダにあるデータの一覧が表示されています。構成はメインウィンドウのブラウザパネルとほぼ同じですが、列が展開出来るようになっています。

列の下位に表示されるのは、基本統計量演算子であり、以下が用意されています。

- 観測数
- 平均
- 標準偏差
- 分散
- 95 % 信頼区間下限
- 95 % 信頼区間上限
- 変動係数
- 最小値
- 25 % 値

- 中央値
- 75 % 値
- 最大値
- レンジ

基本統計量演算子をダブルクリックすると、スクリプト定義エディタ内のカーソル位置に、その統計量を求める式が挿入されます。

また、基本統計量演算子をスクリプト定義エディタ内にドラッグする事での任意のカーソル位置に、その統計量を求める式が挿入する事も出来ます。

一度もモデルを実行していない場合は、出力フォルダが空のため、ブラウザ内には何も表示されません。また、モデルを編集して実行すると出力フォルダに出力されるデータが変わるため、データブラウザに表示されるデータも変わってきます。

5.7 分析スクリプトの実行

分析スクリプトの実行方法についてを説明します。

モデルを開いている状態で、モデル編集画面の部品がない場所で右クリックして表示されるメニュー (3.5.1 章) から「分析スクリプトを開始する」を選ぶとサブメニューが表示されますので、そこから実行したい分析スクリプト名を選ぶと、分析スクリプトを実行を開始します。

モデル実行の結果はモデル出力タブに出力されます。

分析スクリプトは、分析スクリプト編集画面 (5.6 章) で編集する事が出来ます。

分析スクリプト実行中にエラーが発生した場合、モデルエラータブにエラーが表示されます。エラーの修正方法については、5.11 章を参照して下さい。

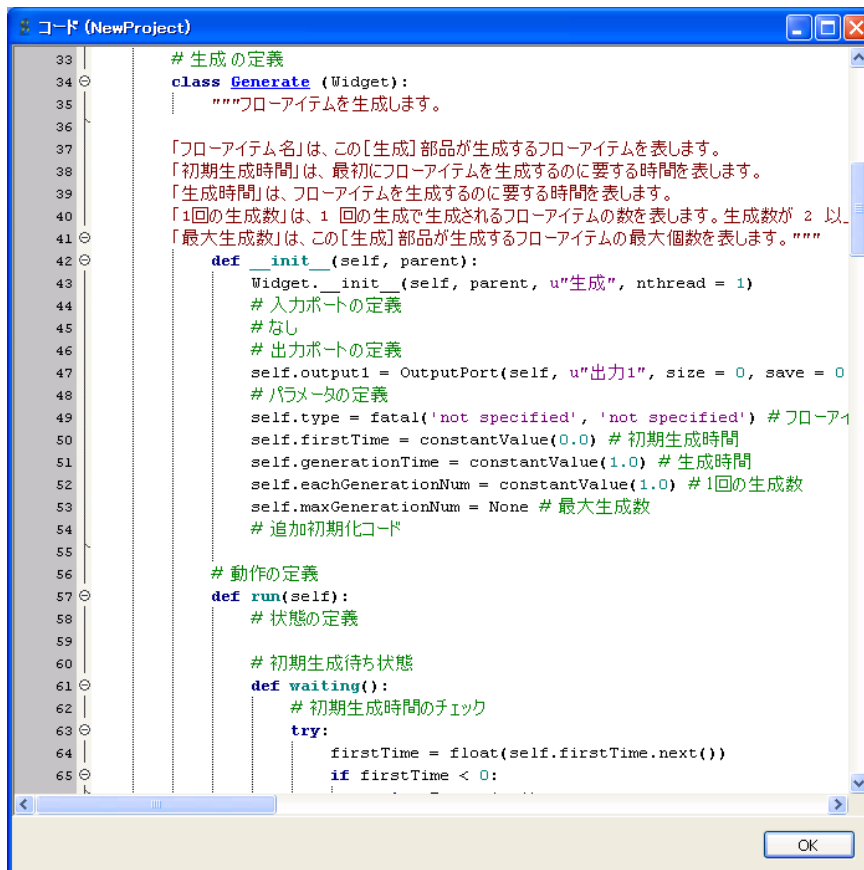
5.8 コード表示

モデルを開いている状態で、モデルメニュー (3.2.3 章) の「コードを表示する」を選択すると自動生成されたコードが表示されます。あるいはツールバーのボタン (3.3.7 章)、モデル編集画面の部品がない場所で右クリックして表示されるメニュー (3.5.1 章) から実行出来ます。

コード表示編集画面 (図 5.26) では、作成したモデルから自動生成されたコードを確認出来ます。

なお、コード表示画面はコードの編集は行えません。部品編集画面のコード編集タブ (6.1.3 章) やカスタムコードの編集 (5.5 章) で編集して下さい。

自動生成コードの構成は以下のようになります。



```
33 # 生成の定義
34 class Generate (Widget):
35     """フローアイテムを生成します。
36
37     「フローアイテム名」は、この[生成]部品が生成するフローアイテムを表します。
38     「初期生成時間」は、最初にフローアイテムを生成するのに要する時間を表します。
39     「生成時間」は、フローアイテムを生成するのに要する時間を表します。
40     「1回の生成数」は、1回の生成で生成されるフローアイテムの数を表します。生成数が2以上
41     「最大生成数」は、この[生成]部品が生成するフローアイテムの最大個数を表します。"""
42     def __init__(self, parent):
43         Widget.__init__(self, parent, u"生成", nthread = 1)
44         # 入力ポートの定義
45         # なし
46         # 出力ポートの定義
47         self.output1 = OutputPort(self, u"出力1", size = 0, save = 0)
48         # パラメータの定義
49         self.type = fatal('not specified', 'not specified') # フローアイテム
50         self.firstTime = constantValue(0.0) # 初期生成時間
51         self.generationTime = constantValue(1.0) # 生成時間
52         self.eachGenerationNum = constantValue(1.0) # 1回の生成数
53         self.maxGenerationNum = None # 最大生成数
54         # 追加初期化コード
55
56     # 動作の定義
57     def run(self):
58         # 状態の定義
59
60         # 初期生成待ち状態
61         def waiting():
62             # 初期生成時間のチェック
63             try:
64                 firstTime = float(self.firstTime.next())
65                 if firstTime < 0:
```

図 5.26: コード表示画面

5.8.1 自動生成コードの構成

自動生成コードの構造は以下のようになります。

```
# -*- Mode: Python; coding: utf-8-dos; -*-
```

ここに初期化コードが挿入されます。

```
# カスタムコード
```

```
...
```

ここにカスタムコード編集画面で編集したコードが挿入されます。

```
# シミュレーションパラメータの定義
```

```
class SimulationParam:
    def __init__(self):
        ...
```

ここにパラメータ編集画面のパラメータタブで定義したパラメータが挿入されます。

```
# シミュレーターの定義
```

```
class Simulator (SimulatorBase):
    def __init__(self,
                 param = SimulationParam(),
                 ...);
```

ここにパラメータ編集画面の基本設定タブで指定した、入力フォルダ、出力フォルダ、シミュレーション終了時間が挿入されます。

```
SimulatorBase.__init__(self, param, inputDir, outputDir, until)
```

```
# アイテム の定義
```

```
class Item (FlowItem):
    """..."""
    def __init__(self):
        FlowItem.__init__(self, name = u"...")
        # パラメータの定義
        ...
    ...
```

ここにモデル上に配置したフローアイテムの定義が挿入されます。

```
# 資源の定義
```

```
...
```

ここにモデル上に配置した資源の定義が挿入されます。

```
# ... の定義
class ... (Widget):
    """..."""
    def __init__(self, parent):
        Widget.__init__(self, parent, u"...", nthread = 1)
        # 入力ポートの定義
        ...
        # 出力ポートの定義
        ...
        # パラメータの定義
        ...
    # 動作の定義
    def run(self):
        ...
    ...
```

ここにモデル上に配置した部品の定義が挿入されます。

ここにモデル上に配置したリアルタイムグラフの定義が挿入されます。

```
# オブジェクトの生成
...
```

ここに部品のインスタンスを生成するコードが挿入されます。

```
# リンクの生成
...
```

ここに部品間のリンクを追加するコードが挿入されます。

```
def run(self):
    # オブジェクトの起動
    ...
```

ここに部品を起動するコードが挿入されます。

```
# グラフの表示
...
```

ここにリアルタイムグラフを描画するコードが挿入されます。ただし、リアルタイムグラフがひとつも存在しない場合は、ここは空になります。

```
# シミュレーション開始
start(until = self.until)
# 結果の保存
self.save()
self.flush()
def objective(self):
    ...
...
```

ここにスクリプト編集画面で定義したスクリプトの定義が挿入されます。

```
if __name__ == "__main__":
    # 実行時の処理
    ...
```

ここにシミュレーションを実行するコードが挿入されます。通常実行、一括実行、感度分析、実験計画、最適化を行うかによってコードは変わります。

5.9 モデルの実行

モデルの実行方法についてを説明します。

モデルを開いている状態で、モデルメニュー (3.2.3 章) の「モデルを開始する」を選択するとモデルの実行を開始します。あるいはツールバーのボタン (3.3.5 章)、モデル編集画面の部品がない場所で右クリックして表示されるメニュー (3.5.1 章) から実行出来ます。

パラメータ編集画面の基本設定タブの実行種類によって、表示される画面が異なります。

なお、並列実行を行うように設定 (3.2.1 章) されている場合、一括実行、感度分析実行、実験計画実行、最適化実行は、出来る限り並列に実行します。並列実行モードの時は、タスクスケジューラウィンドウが表示されます (図 5.27)。

タスクスケジューラウィンドウは各コアの実行状況を示します。第 1 列は各コアの状態、第 2 列は実行しているタスク Id を示しています。このウィンドウは一切操作は出来ません。シミュレーションの実行が停止すると、自動的に閉じます。

5.9.1 通常実行

実行種類が通常実行の場合は、以下のようなモデル実行画面が表示されます。(図 5.28)

	状態	タスクId
1	Running	79
2	Running	77
3	Idle	
4	Idle	

キュー: 0

図 5.27: タスクスケジューラウィンドウ

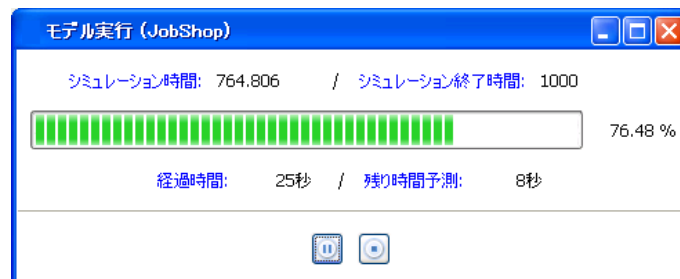


図 5.28: モデル実行画面

モデルの実行が始まると、一時停止、停止ボタンがクリック出来る状態 (実行状態) になります。

実行状態で停止ボタンを押すと、実行を停止し、モデル実行画面を閉じます。

実行状態で一時停止ボタンを押すと、一時停止状態になり、一時停止ボタンが再開ボタンに変わります。

一時停止状態で再開ボタンを押すと、実行状態に戻り、再開ボタンが一時停止ボタンに変わります。

一時停止、あるいは、停止ボタンを押すと、出力フォルダに実行結果が書き込まれます。

シミュレーション時間

モデル内でのシミュレーション時間が表示されます。

シミュレーション終了時間

パラメータ編集画面で設定したシミュレーション終了時間が表示されます。

経過時間

5.9 モデルの実行

シミュレーションを開始してから経過した(実)時間が表示されます。一時停止している期間は計測されません。

残り時間予測

シミュレーション時間の進む速さが一定だとした時の、残りの(実)時間の予測が表示されます。

5.9.2 一括実行

実行種類が一括実行の場合は、以下のようなモデル実行画面が表示されます。(図 5.29)



図 5.29: モデル実行画面 (一括実行)

モデルの実行が始まると、一時停止、停止ボタンがクリック出来る状態(実行状態)になります。

実行状態で停止ボタンを押すと、実行を停止し、モデル実行画面を閉じます。

実行状態で一時停止ボタンを押すと、一時停止状態になり、一時停止ボタンが再開ボタンに変わります。

一時停止状態で再開ボタンを押すと、実行状態に戻り、再開ボタンが一時停止ボタンに変わります。

一時停止、あるいは、停止ボタンを押すと、出力フォルダに実行結果が書き込まれます。

上段には、一括実行全体の経過が表示されます。中段には、各反復（各パラメータの組合せ）の経過が表示されます。下段には、一括実行全体の中の一つの実行単位のシミュレーションの経過が表示されます。

反復回数

実行単位番号が表示されます。

最大反復回数

全てのパラメータの組合せ数か最大反復回数のうち小さい方が表示されます。ただし、パラメータの設定によっては実行してみないと実行回数が分からない場合があり、その場合は見積りが間違える場合があります。また、無限系列が含まれる場合は表示されません。

経過時間 (上段)

シミュレーションを開始してから経過した (実) 時間が表示されます。一時停止している期間は計測されません。

残り時間予測 (上段)

シミュレーション時間の進む速さが一定だとした時の、残りの (実) 時間の予測が表示されます。

レプリケーション番号

実行中の反復におけるレプリケーション番号が表示されます。並列実行時 (3.2.1 章) は、表示されません。

最大レプリケーション番号

実行中の反復における最大レプリケーション番号が表示されます。

経過時間 (中段)

反復を開始してから経過した (実) 時間が表示されます。一時停止している期間は計測されません。

残り時間予測 (中段)

シミュレーション時間の進む速さが一定だとした時の、反復の残りの(実)時間の予測が表示されます。

シミュレーション時間

ひとつの実行単位でのシミュレーション時間が表示されます。並列実行時(3.2.1章)は、表示されません。

シミュレーション終了時間

ひとつの実行単位のシミュレーション終了時間が表示されます。

経過時間(下段)

ひとつの実行単位のシミュレーションを開始してから経過した(実)時間が表示されます。一時停止している期間は計測されません。

残り時間予測(下段)

ひとつの実行単位のシミュレーション時間の進む速さが一定だとした時の、残りの(実)時間の予測が表示されます。

5.9.3 感度分析実行

実行種類が感度分析の場合は、以下のようなモデル実行画面が表示されません。(図 5.30)

モデルの実行が始まると、一時停止、停止ボタンがクリック出来る状態(実行状態)になります。

実行状態で停止ボタンを押すと、実行を停止し、モデル実行画面を閉じます。

実行状態で一時停止ボタンを押すと、一時停止状態になり、一時停止ボタンが再開ボタンに変わります。

一時停止状態で再開ボタンを押すと、実行状態に戻り、再開ボタンが一時停止ボタンに変わります。

一時停止、あるいは、停止ボタンを押すと、出力フォルダに実行結果が書き込まれます。

反復回数

実行終了したパラメータの個数が表示されます。

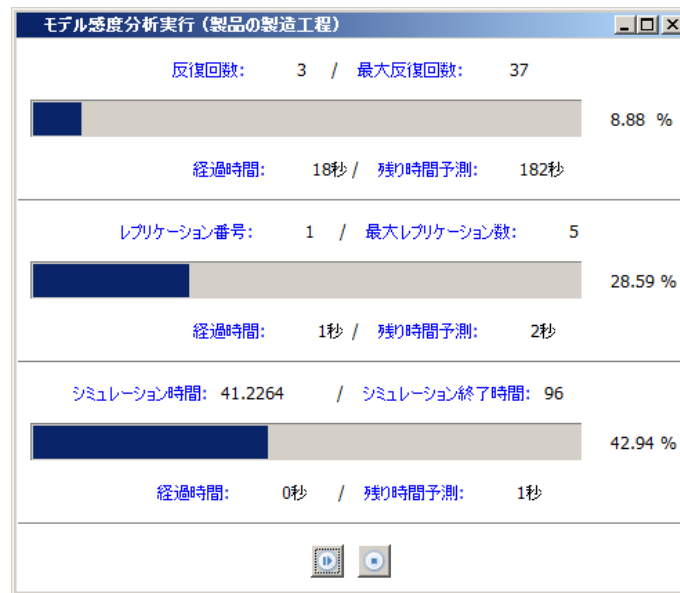


図 5.30: モデル実行画面 (感度分析実行)

最大反復回数

実行対象のパラメータの総数が表示されます。

経過時間 (上段)

シミュレーションを開始してから経過した (実) 時間が表示されます。一時停止している期間は計測されません。

残り時間予測 (上段)

シミュレーション時間の進む速さが一定だとした時の、残りの (実) 時間の予測が表示されます。

レプリケーション番号

実行中の反復におけるレプリケーション番号が表示されます。並列実行時 (3.2.1 章) は、表示されません。

最大レプリケーション番号

実行中の反復における最大レプリケーション番号が表示されます。

経過時間 (中段)

反復を開始してから経過した (実) 時間が表示されます。一時停止している期間は計測されません。

残り時間予測 (中段)

シミュレーション時間の進む速さが一定だとした時の、反復の残りの (実) 時間の予測が表示されます。

シミュレーション時間

ひとつの実行単位でのシミュレーション時間が表示されます。並列実行時 (3.2.1 章) は、表示されません。

シミュレーション終了時間

ひとつの実行単位のシミュレーション終了時間が表示されます。

経過時間 (下段)

ひとつの実行単位のシミュレーションを開始してから経過した (実) 時間が表示されます。一時停止している期間は計測されません。

残り時間予測 (下段)

ひとつの実行単位のシミュレーション時間の進む速さが一定だとした時の、残りの (実) 時間の予測が表示されます。

5.9.4 実験計画実行

実行種類が実験計画の場合は、以下の様なモデル実行画面が表示されます。(図 5.31)

モデルの実行が始まると、一時停止、停止ボタンがクリック出来る状態 (実行状態) になります。

実行状態で停止ボタンを押すと、実行を停止し、モデル実行画面を閉じます。

実行状態で一時停止ボタンを押すと、一時停止状態になり、一時停止ボタンが再開ボタンに変わります。

一時停止状態で再開ボタンを押すと、実行状態に戻り、再開ボタンが一時停止ボタンに変わります。



図 5.31: モデル実行画面 (実験計画実行)

一時停止、あるいは、停止ボタンを押すと、出力フォルダに実行結果が書き込まれます。

反復回数

実行終了したパラメータの個数が表示されます。

最大反復回数

実行対象のパラメータの総数が表示されます。

経過時間 (上段)

シミュレーションを開始してから経過した (実) 時間が表示されます。一時停止している期間は計測されません。

残り時間予測 (上段)

シミュレーション時間の進む速度が一定だとした時の、残りの (実) 時間の予測が表示されます。

シミュレーション時間

ひとつの実行単位でのシミュレーション時間が表示されます。並列実行時 (3.2.1 章) は、表示されません。

シミュレーション終了時間

ひとつの実行単位のシミュレーション終了時間が表示されます。

経過時間 (下段)

ひとつの実行単位のシミュレーションを開始してから経過した (実) 時間が表示されます。一時停止している期間は計測されません。

残り時間予測 (下段)

ひとつの実行単位のシミュレーション時間の進む速さが一定だとした時の、残りの (実) 時間の予測が表示されます。

5.9.5 最適化実行

実行種類が最適化の場合は、以下のようなモデル実行画面が表示されます。
(図 5.32)

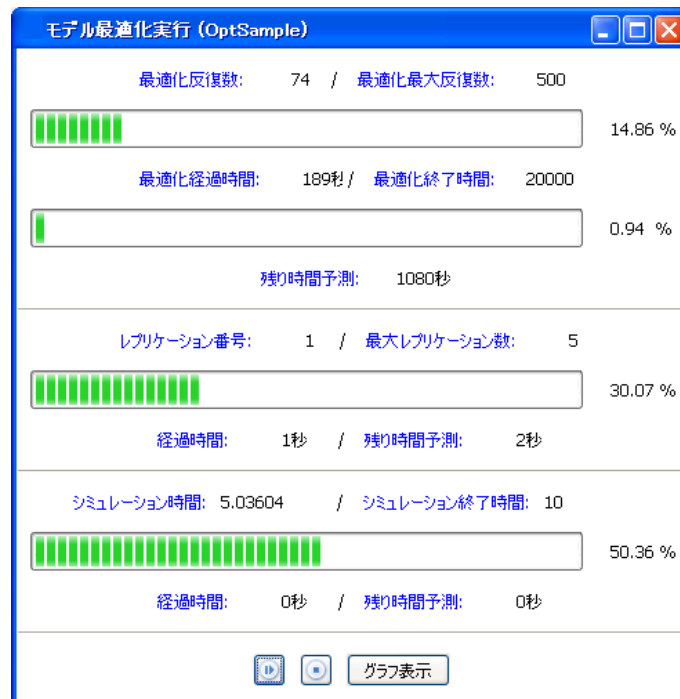


図 5.32: モデル実行画面 (最適化)

モデルの実行が始まると、一時停止、停止ボタンがクリック出来る状態 (実行状態) になります。

実行状態で停止ボタンを押すと、実行を停止し、モデル実行画面を閉じます。

実行状態で一時停止ボタンを押すと、一時停止状態になり、一時停止ボタンが再開ボタンに変わります。

一時停止状態で再開ボタンを押すと、実行状態に戻り、再開ボタンが一時停止ボタンに変わります。

一時停止、あるいは、停止ボタンを押すと、出力フォルダに実行結果が書き込まれます。

グラフ表示ボタンを押すと、最適化の収束状況を確認する事が出来ます。
(5.9.5 章)

最適化反復数

現在の反復数が表示されます。

最適化最大反復数

最大反復数が表示されます。

最適化経過時間

最適化を開始してから経過した (実) 時間が表示されます。一時停止している期間は計測されません。

最適化終了時間

最適化を停止する時間が表示されます。

レプリケーション番号

実行中の反復におけるレプリケーション番号が表示されます。並列実行時 (3.2.1 章) は、表示されません。

最大レプリケーション番号

実行中の反復における最大レプリケーション番号が表示されます。

経過時間 (上部)

反復を開始してから経過した (実) 時間が表示されます。一時停止している期間は計測されません。

残り時間予測 (上部)

シミュレーション時間の進む速さが一定だとした時の、反復の残りの (実) 時間の予測が表示されます。

シミュレーション時間

ひとつの実行単位でのシミュレーション時間が表示されます。並列実行時 (3.2.1 章) は、表示されません。

シミュレーション終了時間

ひとつの実行単位のシミュレーション終了時間が表示されます。

経過時間 (下部)

ひとつの実行単位のシミュレーションを開始してから経過した (実) 時間が表示されます。一時停止している期間は計測されません。

残り時間予測 (下部)

ひとつの実行単位のシミュレーション時間の進む速さが一定だとした時の、残りの (実) 時間の予測が表示されます。

最適化進捗グラフ

最適化進捗グラフは単目的の場合 (DFO、PSO) と多目的 (MOPSO) の場合で異なるグラフが表示されます。

単目的の場合、最適化進捗グラフは以下のような構成になっています。(図 5.33)

2 段の表示になっており、横軸は反復回数を示します。

上段は、目的関数の推移を示しています。小さな点は各レプリケーションの標本値を示しています。大きな点は、その反復におけるレプリケーションの平均を示しています。赤線は、その時点での最適な解を示しています。

下段は、最適化変数の推移を示しています。各反復において、試行する変数の値が 0 から 1 に規格化されてプロットされます。

多目的の場合、最適化進捗グラフは以下のような構成になっています。(図 5.34)

それぞれの目的変数に対する対散布図になっており、パレート解の様子を表しています。

第 5 章 モデル

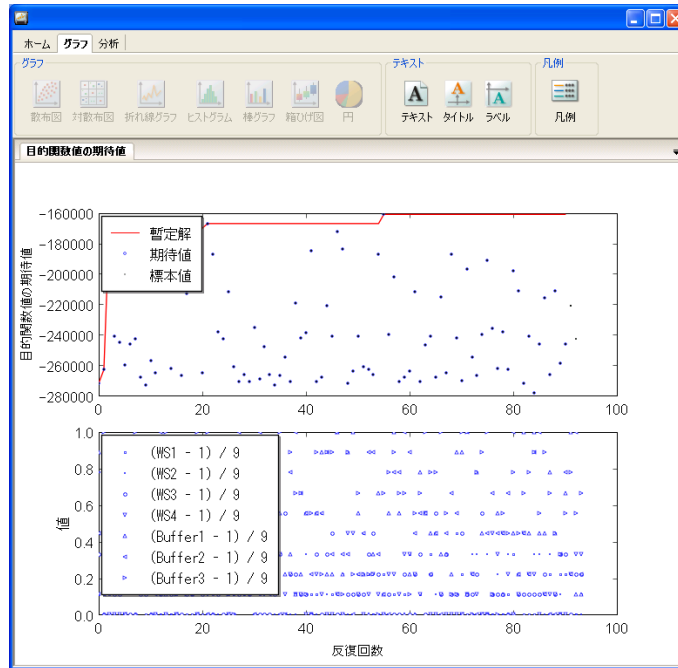


図 5.33: 最適化進捗グラフ (単目的)

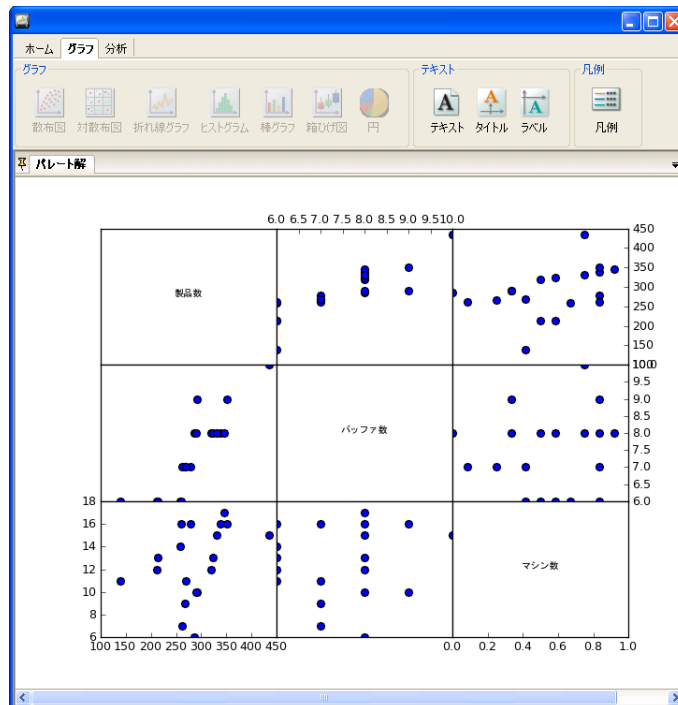


図 5.34: 最適化進捗グラフ (多目的)

5.10 実行結果

5.10.1 通常実行

通常実行を停止あるいは実行完了すると、モデル出力タブに、objective()の結果が表示されます。

通常実行を一時停止、停止、あるいは実行完了すると、出力フォルダに結果が出力されます。その結果は、サマリ表示 (23.1 章) や、データの表示 (23.2 章) で確認する事が出来ます。また、独自に定義した分析スクリプト (5.7 章) で統計量を求める事も出来ます。

実行中にエラーが発生した場合、モデルエラータブにエラーが表示されます。エラーの修正方法については、5.11 章を参照して下さい。

5.10.2 一括実行

一括実行を一時停止、停止、あるいは実行完了すると、出力フォルダに各実行単位の結果が出力されます。5.4 章の一括実行タブ (5.4.3) の「結果を全て保存する」にチェックが入っている場合は、実行を完了した全ての実行単位の結果が出力されます。チェックが入っていない場合は、最後の実行単位の結果のみ出力されます。また、実行単位の結果以外に「一括実行結果」と「一括実行詳細結果」という名前のモニターも出力されます。

一括実行結果モニターは、5.4 章の一括実行タブで設定した「パラメータ」と「出力」を含む形式で、図 5.35 のような結果が出力されます。ただし、複数のレプリケーションを行った場合、出力列の値は平均値となります。

パラメータ名 ₁	...	パラメータ名 _n	出力名 ₁	...	出力名 _m
パラメータ値 ₁₁	...	パラメータ値 _{1n}	出力値 ₁₁	...	出力値 _{1m}
⋮	⋱	⋮	⋮	⋱	⋮
パラメータ値 _{k1}	...	パラメータ値 _{kn}	出力値 _{k1}	...	出力値 _{km}

図 5.35: 一括実行結果モニター

一括実行詳細結果モニターは一括実行結果モニターの内容をより詳しくしたもので、図 5.36 のような形式で出力されます。ここで、反復回数は何番目に実行したパラメータの組合せかを表す番号で、レプリケーション番号は何回目のレプリケーションかを表す番号です。反復回数が NNN でレプリケーション番号が RRR の行は、出力フォルダにある「出力フォルダ名-itrr-NNN-RRR」のフォルダの実行単位の結果と対応しています。

一括実行結果と一括実行詳細結果はデータの表示 (23.2 章) で確認する事が出来ます。

反復回数	レプリケーション番号	パラメータ名 ₁	...	パラメータ名 _n	出力名 ₁	...	出力名 _m
反復回数 ₁	レプリケーション番号 ₁	パラメータ値 ₁₁	...	パラメータ値 _{1n}	出力値 ₁₁	...	出力値 _{1m}
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
反復回数 _k	レプリケーション番号 _k	パラメータ値 _{k1}	...	パラメータ値 _{kn}	出力値 _{k1}	...	出力値 _{km}

図 5.36: 一括実行詳細結果モニター

一括実行において、ある実行単位の実行中にエラーが発生した場合、モデルエラータブにその実行単位のパラメータと、エラー情報が表示されますが、その次の実行単位に実行を進めます。一括実行結果モニターと一括実行詳細結果モニターにはエラーが発生した実行単位の結果は含まれません。エラーの修正方法については、5.11 章を参照して下さい。

5.10.3 感度分析

感度分析実行を一時停止、停止、あるいは実行完了すると、出力フォルダに各実行単位の結果が出力されます。5.4 章の感度分析タブ (5.4.4) の「結果を全て保存する」にチェックが入っている場合は、実行を完了した全ての実行単位の結果が出力されます。チェックが入っていない場合は、最後の実行単位の結果のみ出力されます。また、実行単位の結果以外に、「感度分析結果」、「感度分析詳細結果」と、パラメータごとの「感度分析結果 - パラメータ名」という名前のモニターも出力されます。

感度分析結果モニターは、5.4 章の感度分析タブで設定した「パラメータ」と「出力」から定まる、図 5.37 のような列を持ちます。

感度分析詳細結果モニターは、レプリケーションを含む各実行結果をまとめたものになっており、図 5.38 のような形式で出力されます。ここで、反復回数は何番目に実行したパラメータの組合せかを表す番号で、レプリケーション番号は何回目のレプリケーションかを表す番号です。反復回数が NNN でレプリケーション番号が RRR の行は、出力フォルダにある「出力フォルダ名-itr-NNN-RRR」のフォルダの実行単位の結果と対応しています。

パラメータごとの「感度分析結果 - パラメータ名」モニターは、パラメータの基準値に対する変化と出力をまとめたものになっており、図 5.39 のような形式で出力されます。ここで、変化率は基準値からの変化率を表しています。

5.10.4 実験計画

実験計画実行を一時停止、停止、あるいは実行完了すると、出力フォルダに各実行単位の結果が出力されます。5.4 章の実験計画タブ (5.4.5) の「結果を全て保存する」にチェックが入っている場合は、実行を完了した全ての実行単位の結果が出力されます。チェックが入っていない場合は、最後の実行

5.10 実行結果

列名	説明
パラメータ名	パラメータの名称です
最大減少率 (%)	パラメータの基準値に対する最小値の変化割合を表します
最小値	パラメータの最小値です
基準値	パラメータの基準値です
最大増加率 (%)	パラメータの基準値に対する最大値の変化割合を表します
最大値	パラメータの最大値です
出力 ₁ 期待値最小値	出力 ₁ の期待値の最小値です
出力 ₁ 期待値基準値	基準値における出力 ₁ の期待値です
出力 ₁ 期待値最大値	出力 ₁ の期待値の最大値です
出力 ₁ 微分係数推定値	出力 ₁ の基準値周りの微分係数の推定値です
⋮	⋮
出力 _k 期待値最小値	出力 _k の期待値の最小値です
出力 _k 期待値基準値	基準値における出力 _k の期待値です
出力 _k 期待値最大値	出力 _k の期待値の最大値です
出力 _k 微分係数推定値	出力 _k の基準値周りの微分係数の推定値です

図 5.37: 感度分析結果モニターの列

反復回数	レプリケーション番号	パラメータ名 ₁	⋯	パラメータ名 _n	出力名 ₁	⋯	出力名 _m
反復回数 ₁	レプリケーション番号 ₁	パラメータ値 ₁₁	⋯	パラメータ値 _{1n}	出力値 ₁₁	⋯	出力値 _{1m}
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
反復回数 _k	レプリケーション番号 _k	パラメータ値 _{k1}	⋯	パラメータ値 _{kn}	出力値 _{k1}	⋯	出力値 _{km}

図 5.38: 感度分析詳細結果モニター

パラメータ値	変化率	出力名 ₁	⋯	出力名 _m
パラメータ値 ₁	変化率 ₁	出力値 ₁₁	⋯	出力値 _{1m}
⋮	⋮	⋮	⋮	⋮
パラメータ値 _k	変化率 _k	出力値 _{k1}	⋯	出力値 _{km}

図 5.39: 「感度分析結果 - パラメータ名」モニター

単位の結果のみ出力されます。また、実行単位の結果以外に「実験計画実行結果」という名前のモニターも出力されます。

実験計画実行結果モニターは 5.4 章の実験計画タブで設定した、「使用する計画」設定欄のモニターに含まれるパラメータと、「出力」を含む形式で、図 5.40 のような形式で出力されます。ここで、反復回数は何番目に実行したパラメータの組合せかを表す番号です。反復回数が NNN の行は出力フォルダにある「出力フォルダ名-itrr-NNN」のフォルダの実行単位の結果と対応しています。

反復回数	パラメータ名 ₁	...	パラメータ名 _n	出力名 ₁	...	出力名 _m
反復回数 ₁	パラメータ値 ₁₁	...	パラメータ値 _{1n}	出力値 ₁₁	...	出力値 _{1m}
⋮	⋮	⋮	⋮	⋮	⋮	⋮
反復回数 _k	パラメータ値 _{k1}	...	パラメータ値 _{kn}	出力値 _{k1}	...	出力値 _{km}

図 5.40: 実験計画実行結果モニター

実験計画実行結果はデータの表示 (23.2 章) で確認する事が出来ます。

実験計画実行において、ある実行単位の実行中にエラーが発生した場合、モデルエラータブにその実行単位のパラメータと、エラー情報が表示されますが、その次の実行単位に実行を進めます。実験計画実行結果モニターにはエラーが発生した実行単位の結果は含まれません。エラーの修正方法については、5.11 章を参照して下さい。

5.10.5 最適化

最適化を一時停止、停止、あるいは実行完了すると、出力フォルダには最後に実行している実行単位の結果に加えて、「最適化実行結果」、「最適化実行詳細結果」という名前のモニターが出力されます。多目的最適化を行っている場合には、更に「パレート解」という名前のモニターが出力されます。

最適化実行結果モニターは、図 5.41 のような結果が形式で出力されます。

パラメータ名 ₁	...	パラメータ名 _n	目的関数期待値
パラメータ値 ₁₁	...	パラメータ値 _{1n}	期待値 ₁
⋮	⋮	⋮	⋮
パラメータ値 _{k1}	...	パラメータ値 _{kn}	期待値 _k

図 5.41: 最適化実行結果モニター

各行は反復の結果を示します。各反復におけるパラメータと、その時の目的関数の期待値を示しています。多目的最適化の場合には複数の目的関数値が出力されます。

5.11 モデルのデバッグ

最適化実行詳細結果モニターは、図 5.42 のような結果が形式で出力されます。

反復回数	レプリケーション番号	パラメータ名 ₁	...	パラメータ名 _n	目的関数値
反復回数 ₁	レプリケーション番号 ₁	パラメータ値 ₁₁	...	パラメータ値 _{1n}	目的関数値 ₁
⋮	⋮	⋮	⋮	⋮	⋮
反復回数 _k	レプリケーション番号 _m	パラメータ値 _{k1}	...	パラメータ値 _{kn}	目的関数値 _k

図 5.42: 最適化実行詳細結果モニター

各行はレプリケーションの結果を示します。各レプリケーションにおけるパラメータと、その時の目的関数値を示しています。多目的最適化の場合には複数の目的関数値が出力されます。

パレート解のモニターは、図 5.43 のような結果が形式で出力されます。

パラメータ名 ₁	...	パラメータ名 _n	目的関数期待値 ₁	...	目的関数期待値 _m
パラメータ値 ₁₁	...	パラメータ値 _{1n}	期待値 ₁₁	...	期待値 _{1m}
⋮	⋮	⋮	⋮	⋮	⋮
パラメータ値 _{k1}	...	パラメータ値 _{kn}	期待値 _{k1}	...	期待値 _{km}

図 5.43: パレート解モニター

得られた解のうち、パレート解のみを示します。

5.11 モデルのデバッグ

モデルのデバッグの方法についてを説明します。

5.11.1 部品エラー

モデルを実行すると、モデル出力タブに赤字でエラーが表示される事があります。

[生成] 部品エラー : 1 回の生成数が 0 より大きくありません

このようなエラーが表示された場合 “[、”] で囲まれた名前の部品に設定に誤った設定が含まれています。表示された部品の部品編集画面を開き、エラーを修正して下さい。

5.11.2 部品警告

モデルを実行すると、以下のような警告が表示される事があります。

[生成] 部品警告 : 最大生成数が 0 より大きくありません。最大生成数を無限とみなします。

このような警告が表示された場合 “[、”]” で囲まれた名前の部品に誤った設定が含まれている可能性があります。意図してそのような設定を行っている場合もありますが、大抵の場合は誤った設定です。表示された部品の部品編集画面を開き、適宜修正して下さい。

5.11.3 その他のエラー

モデルを実行すると、モデル出力タブに、赤字で以下のメッセージが表示される事があります。

[エラー出力があります]

その場合は、モデルエラータブを開くと詳細情報を確認する事が出来ます。モデルエラータブには以下のようなエラーが表示されている場合があります。

モデル ... の開始: ...

Exception in thread Thread-1:

Traceback (most recent call last):

...

Project ..., line ..., in ...

TypeError: unsupported operand type(s) for /: 'instance' and 'int'

モデル ... の終了: ...

その中で、Project ... が含まれている場合があります。

Project ..., line XXX, in ...

この場合は、自動生成コード上の XXX 行でエラーが発生した事を示しています。

その行をクリックすると、自動生成コードが表示され、エラー行が表示されます。あるいは、コード表示 (5.8 章) を行い、エラー発生行を参照しても構いません。エラー発生行を起点に上部のコードを参照すると、

```
class ... (Widget):
```

という部品のクラス定義があると思います。その部品内で問題が発生していますので、部品を開き、適宜問題点を修正して下さい。

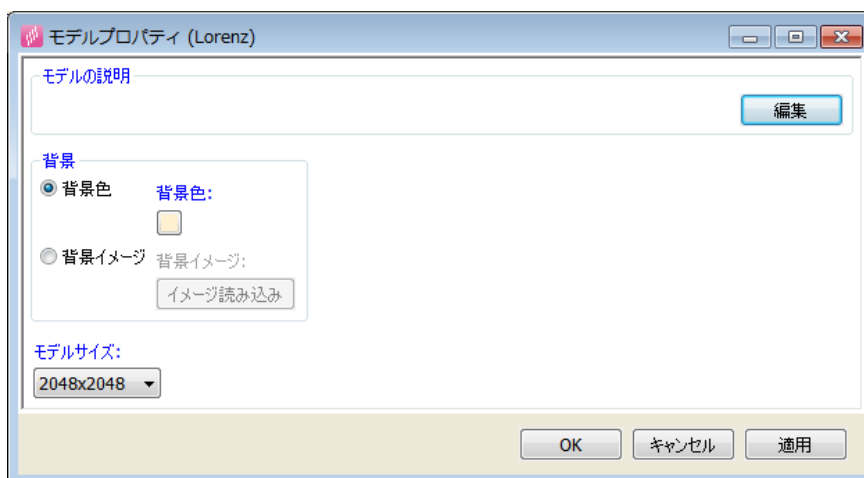


図 5.44: モデルプロパティ編集

5.12 モデルプロパティ編集

モデルを開いている状態で、モデルメニュー (3.2.3 章) の「モデルプロパティを編集する」を選択するとモデルプロパティ編集画面 (図 5.44) が表示されます。

モデルの説明

このモデルの説明を自由に記述できます

背景

このモデルの背景色、あるいは、背景イメージを設定します。

モデルサイズ

このモデルのサイズを、1024x1024, 2048x2048, 4096x4096 の中から選択できます。

5.13 ナビゲーションウィンドウ

モデルを開いている状態で、モデルメニュー (3.2.3 章) の「ナビゲーションウィンドウを表示する」を選択するとナビゲーションウィンドウ (図 5.45) が表示されます。

ナビゲーションウィンドウの中央のナビゲーションイメージには、現在編集中のモデルの全体が表示されています。中央の明るい部分が、現在モデル

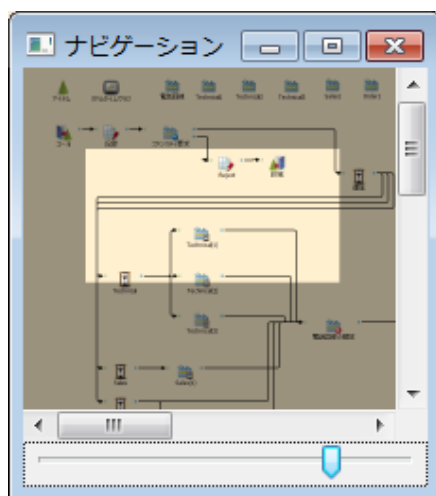


図 5.45: ナビゲーションウィンドウ

編集画面に表示されている領域です。この領域をドラッグする事によって、モデル編集画面上の表示領域を移動させる事ができます。

ナビゲーションイメージの表示領域は、ナビゲーションイメージの右側及び下側にあるスクロールバーによって移動させる事ができます。

ナビゲーションウィンドウの下部にはズームコントロールがあります。ズームコントロールを右に動かすと、ナビゲーションイメージの表示倍率が拡大されます。ズームコントロールを左に動かすと、ナビゲーションイメージの表示倍率が縮小されます。

5.14 Python 拡張環境を編集する

モデルを開いている状態で、モデルメニュー (3.2.3 章) の「Python 拡張環境を編集する」を選択すると Python 環境画面 (図 5.46) が表示されます。

S⁴ Simulation System で作成されたプロジェクトは `python` のコードに変換されて、実行されます。本メニューより、プロジェクト毎に、独自の `python` ライブラリを組み込むことが出来ます。

Python には、PyPI (Python Package Index) と呼ばれる Python のサードパーティライブラリを管理しているサイトがあります。(https://pypi.org/)

本機能は、PyPI からライブラリをインストールするコマンド `pip` のフロントエンドとなっています。(https://pip.pypa.io/)

なお、PyPI の Python ライブラリは、第三者が管理しているものであり、新規のライブラリを導入した場合や、既存のライブラリをバージョンアップした場合、プロジェクトの動作が異ってしまう場合がございます。例えば、実行するとエラーが出る場合や、以前と結果が異ってしまう場合などがござ



図 5.46: Python 環境

います。そのような時は、手動にて、ソースコードの変更、あるいは、基の状態に戻して頂く必要がございます。

Python 環境拡張

Python 環境拡張を利用するか否かを設定します

パッケージ追加

左側のテキストボックスに入力された Python ライブラリをインポートします。

パッケージ一覧については、上記の PyPI のサイトを参照下さい。

なお、バージョン指定でインポートしたい場合は、「パッケージ名==バージョン番号」という文字列を指定する事も可能です。

追加 pip オプション

pip に追加するオプション文字列を設定します。

Wheel を追加

whl ファイルを追加します。ファイルダイアログにて whl ファイルを指定下さい

パッケージ追加にて Python ライブラリ をインポートすると、導入時にエラーが出る場合や、実行時にエラーが出る事があります。

そのような場合、

<https://www.lfd.uci.edu/~gohlke/pythonlibs/>

などで用意されている、whl ファイルを導入すると解決する事があります。

requirements.txt を作成

現在のパッケージの選択状況を requirements.txt に出力します。

requirements.txt に従い更新

requirements.txt に従い、パッケージを選択します。

Python シェルを起動 (拡張環境)

拡張された Python 環境にて Python シェル を起動します。

5.14.1 パッケージリスト

パッケージ

パッケージ名です。

Original

S⁴ Simulation System の標準のバージョンです。標準ではインストールされていない場合は nan になります。

Current

現在拡張環境にインストールされているバージョンです。

更新

このパッケージをを最新のバージョンに更新します。

削除

このパッケージを削除します。

5.15 ソースコードをエクスポートする

ソースコードを単一の Python コードとして外部にエクスポートします。エクスポートされたコードは、編集後、インポートできます。

出力されたコードには、以下のような領域が埋め込まれます。

コード例

```
#region Generate.run
...
任意のユーザコード
...
#endregion Generate.run
```

region で囲まれた領域は、VS code などでは、伸縮可能になります。(外部エディタが対応している場合)

region で囲まれた領域を任意のテキストエディタで編集した後、インポートすると、修正したコードが元のプロジェクトに反映されます。

5.16 ソースコードをインポートする

エクスポートされたコードは、編集後、インポートできます。
インポートファイルを指定すると、確認画面 (図 5.47) が表示されます。
「OK」を押すと、変更が現プロジェクトに反映されます。
状態には以下があります。

変更あり

変更が検知されました。確認の上、OK を押すと変更が現プロジェクトに反映されます。

変更なし

変更が検知されませんでした。OK を押しても変更は現プロジェクトに反映されません。

なお、以下の状態が表示された何らかの問題がある可能性があります。

反映先がありません

ソースコード上には存在するブロックが、現プロジェクト上にありません。OK を押しても変更は現プロジェクトに反映されません。

ソースコード上に存在しない

現プロジェクト上に存在するブロックが、ソースコード上には存在しません。OK を押しても変更は現プロジェクトに反映されません。

5.16 ソースコードをインポートする

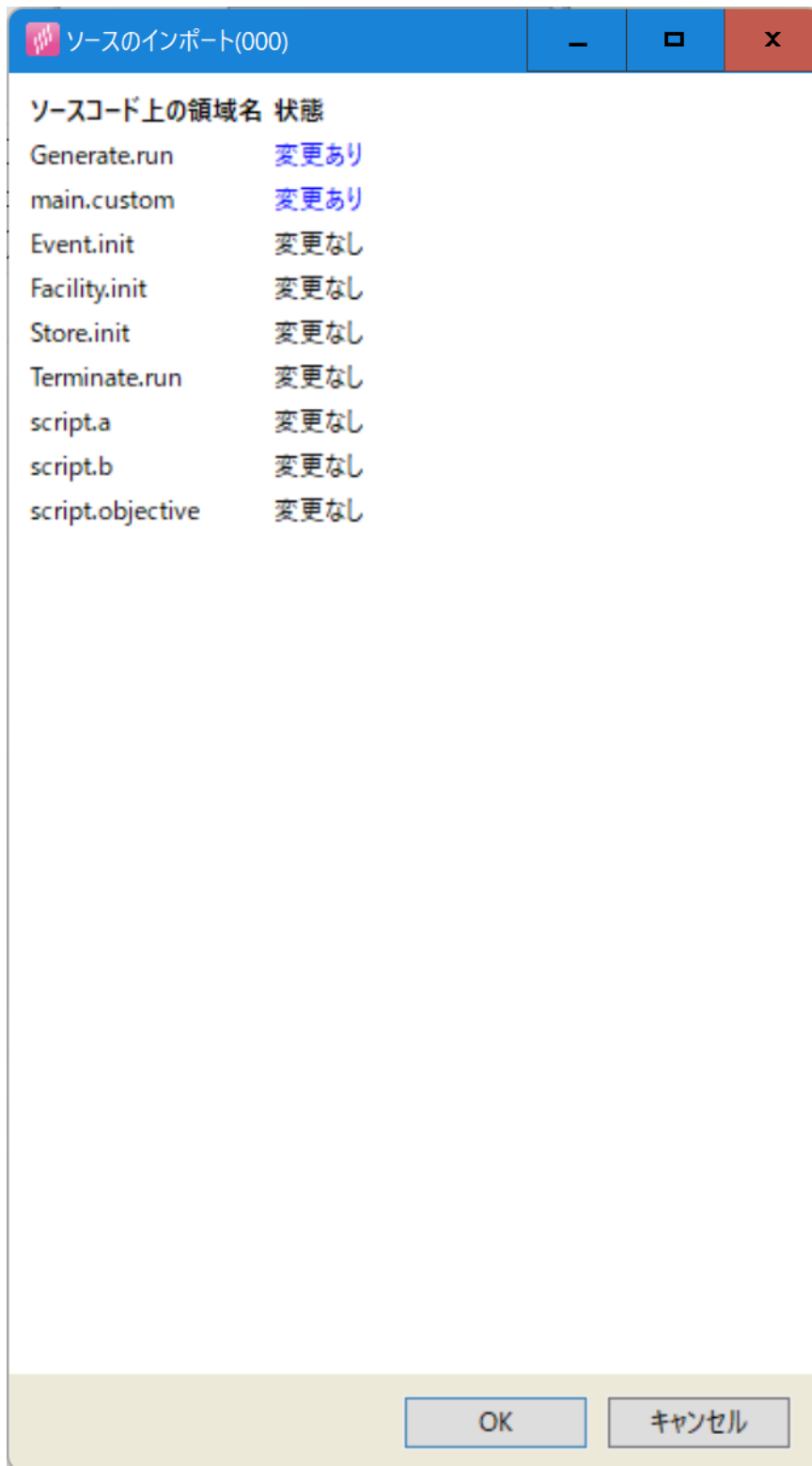


図 5.47: ソースコードインポート

第6章 部品

モデル上に配置する部品についてを説明します。

部品とは、自律的あるいは入力を機に何らかの動作を行い、場合によっては出力を行うものです。任意個の入力ポートと出力ポートを持ちます。

6.1 部品の編集

部品の下のラベルの以外の場所を左ダブルクリック、あるいは、部品の配置されている場所を右クリックして表示されるメニュー (3.5.1 章) から「編集」を選ぶと、部品編集画面が表示されます。

部品編集画面は、以下のタブで構成されます。

6.1.1 共通設定タブ

共通設定タブは 図 6.1 のような画面構成になります。

全ての部品で共通の属性を編集出来ます。

クラス名

自動生成コード (5.8.1 章) におけるこの部品のクラス名になります。通常は編集する必要はありません。

オブジェクト名

自動生成コード (5.8.1 章) におけるこの部品のインスタンス名になります。通常は編集する必要はありません。

説明

モデル編集画面におけるこの部品の名前になります。5.3.15 章の方法でも変更する事が出来ます。

入力以下には入力ポートが表示され、入力ポートの追加・削除、および、各入力ポートを編集する事が出来ます。

6.1 部品の編集

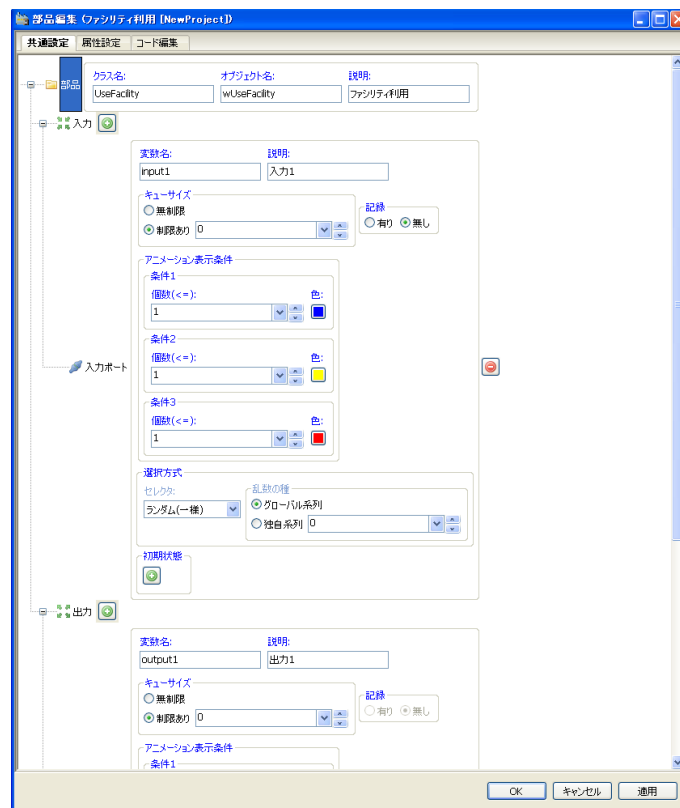


図 6.1: 部品共通設定タブ

出力以下には出力ポートが表示され、出力ポートの追加・削除、および、各出力ポートを編集する事が出来ます。

「+」を押すと、入/出力ポートが追加され、末尾に表示されます。

「-」を押すと、その入/出力ポートが削除されます。

各入/出力ポートは、以下で構成されます。

変数名

そのポートの自動生成コード (5.8.1 章) におけるメンバー名になります。

説明

そのポートの説明になります。

キューサイズ

そのポートのキューの容量になります。

アニメーション表示条件、アニメーション表示範囲

アニメーションに関するパラメータです。詳細は (26 章) アニメーションにあります。

記録

そのポートにキューがある時、キュー変化を記録するかどうかを設定します。記録すると、追加待ち、追加待ち合計、取得待ち、取得待ち合計、バッファで構成される時系列モニターが、「部品名-ポート名」という名前で作成されます。追加待ちと追加待ち合計は常に同じで、取得待ちと取得待ち合計も常に同じになります。追加待ち合計点取得待ち合計はそれぞれ入力側・出力側に並んでいるフローアイテム数です。バッファはポートのキュー内のフローアイテム数です。

選択方式

そのポートにキューがない時、セレクトを設定します。

ポートの動作については、7 章 を参照して下さい。

初期状態

シミュレーション起動時にそのポートのキューに予め並んでいるフローアイテムがある場合、ここで設定する事が出来ます。フローアイテムには追加するフローアイテムの種類を指定します。複製数に指定した数だけフローアイテムが複製されます (deep copy)。通常フローアイテムの各属性は自動的に設定されますが、明示的に指定したい場合は、属性リストで指定する事が出来ます。

6.1.2 属性設定タブ

属性設定タブは 図 6.2 のような画面構成になります。



図 6.2: 部品属性設定タブ

上のヘルプボタンを押すと、その部品の説明が表示されます。部品ごとに表示される属性は変わります。部品の振る舞いを定めます。

6.1.3 コード編集タブ

コード編集タブは、図 6.3 のような画面構成になります。

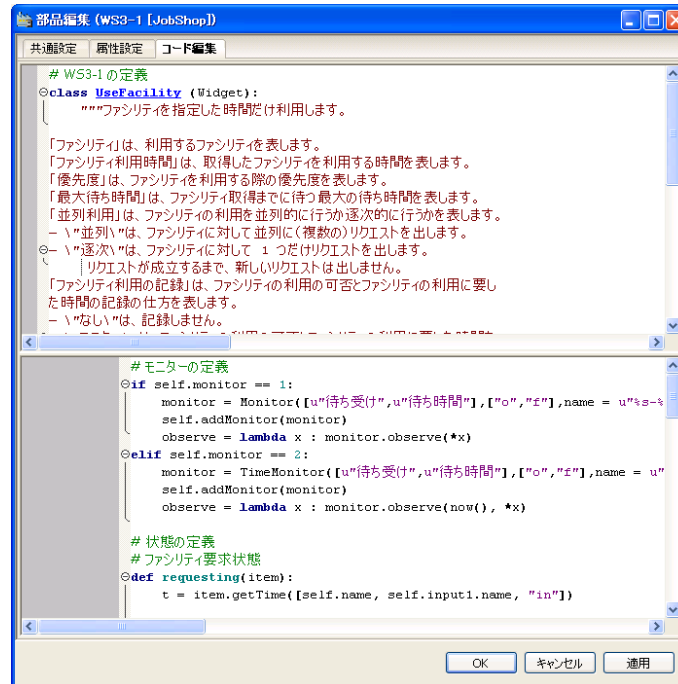


図 6.3: 部品属性設定タブ

上半分が属性設定タブで設定した値により自動生成されたコードです。

下半分がその部品の動作を psim 言語で記述されています。配置した状態では部品ごとの動作が表示されています。自由に編集する事が出来ます。

6.2 生成 (Generate)



図 6.4: 生成

部品「生成」(図 6.4) はフローアイテムを生成します。

フローアイテム名

この [生成] 部品が生成するフローアイテムを表します。

初期生成時間

6.3 終端 (Terminate)

最初にフローアイテムを生成するのに要する時間を表します。

生成時間

フローアイテムを生成するのに要する時間を表します。

生成時間系列の生成

生成時間系列の生成方法を表します。

1 回の生成数

1 回の生成で生成されるフローアイテムの数を表します。生成数が 2 以上の場合はフローアイテムリストとして結合します。フローアイテムリストは部品「分割」(6.15 章)で分割可能です。

最大生成数

この [生成] 部品が生成するフローアイテムの最大個数を表します。

6.3 終端 (Terminate)



図 6.5: 終端

部品「終端」(図 6.5) はフローの終端です。出力はなく、入力されたフローアイテムは破棄されます。

待ち時間

フローアイテムを破棄してから再びフローアイテムを受け取るまでの最短時間を表します。



図 6.6: イベント待ち

6.4 イベント待ち (WaitEvent)

部品「イベント待ち」(図 6.6) は指定したイベント (8.2 章) が発火するまで待ち受けを行います。部品「イベント発火」(6.5 章) と対になる部品です。

イベントリスト

発火を待つイベントを表します。

待ち受け方法

イベントの待ち受け方法を表します。

- "AND" は、順番に関わらず全てのイベントが発火した場合に待ち受けを終了します。
- "OR" は、順番に関わらずいずれかのイベントが発火した場合に待ち受けを終了します。
- "SEQUENCE" は、指定したイベントが指定した順に発火した場合に待ち受けを終了します。

優先度

イベントを待ち受けする際の優先度を表します。

最大待ち時間

イベントを待ち受けする最大の時間を表します。

並列利用

イベント待ち受けを並列的に行うか逐次的に行うかを表します。

- "並列" は、アイテムのイベント待ち受けを並列に行います。
- "逐次" は、アイテムのイベント待ち受けを 1 つだけ行います。イベントが成立するまで、次のアイテムのイベント待ち受けを行いません。

イベントの記録

待ち受けの可否と待ち時間を記録の仕方を表します。

- ”なし”は、記録しません。
- ”モニター”は、待ち受けの可否と待ち時間を記録します。
- ”時系列モニター”は、待ち受けの可否と待ち時間を時系列で記録します。

6.5 イベント発火 (SignalEvent)



図 6.7: イベント発火

部品「イベント発火」(図 6.7) は指定したイベント (8.2 章) を発火します。部品「イベント待ち」(6.4 章) と対になる部品です。

イベントリスト

発火させるイベントを表します。

待ち時間

アイテム受け取り後からイベントを発火させるまでの時間を表します。

発火数

発火させるイベント数を表します。

6.6 ファシリティ要求 (RequestFacility)



図 6.8: ファシリティ要求

部品「ファシリティ要求」(図 6.8) はファシリティ(8.3 章) の利用を要求します。部品「ファシリティ解放」(6.7 章) と対になる部品です。

ファシリティリスト

利用するファシリティを表します。

待ち受け方法

ファシリティの待ち受け方法を表します。

- "AND" は、順番に関わらず全てのファシリティに要求を行ないます。
- "OR" は、順番に関わらずいずれかのファシリティに要求を行ないます。
- "SEQUENCE" は、指定したファシリティに指定した順に要求を行ないます。

優先度

ファシリティを利用する際の優先度を表します。

最大待ち時間

ファシリティ取得までに待つ最大の待ち時間を表します。

並列利用

ファシリティの利用を並列的に行うか逐次的に行うかを表します。

- "並列" は、ファシリティに対して並列に（複数の）要求を出します。
- "逐次" は、ファシリティに対して 1 つだけ要求を出します。要求が成立するまで、新しい要求は出しません。

ファシリティ要求の記録

ファシリティの要求の可否とファシリティの要求に要した時間の記録の仕方を表します。

- "なし" は、記録しません。
- "モニター" は、ファシリティの要求の可否とファシリティの要求に要した時間を記録します。
- "時系列モニター" は、ファシリティの要求の可否とファシリティの要求に要した時間を時系列的に記録します。



図 6.9: ファシリティ解放

6.7 ファシリティ解放 (ReleaseFacility)

部品「ファシリティ解放」(図 6.9) は利用中のファシリティ(8.3 章) を解放します。部品「ファシリティ要求」(6.6 章) と対になる部品です。

ファシリティリスト

解放するファシリティを表します。

6.8 ファシリティ利用 (UseFacility)



図 6.10: ファシリティ利用

部品「ファシリティ利用」(図 6.10) はファシリティ(8.3 章) を指定した時間だけ利用します。部品「ファシリティ要求」(6.6 章) と部品「ファシリティ解放」(6.7 章) の間に部品「遅延」(6.18 章) が挟まっているような動作をします。

ファシリティリスト

利用するファシリティを表します。

待ち受け方法

ファシリティの待ち受け方法を表します。

- "AND" は、順番に関わらず全てのファシリティに要求を行ないます。
- "OR" は、順番に関わらずいずれかのファシリティに要求を行ないます。
- "SEQUENCE" は、指定したファシリティに指定した順に要求を行ないます。

ファシリティ利用時間

取得したファシリティを利用する時間を表します。

優先度

ファシリティを利用する際の優先度を表します。

最大待ち時間

ファシリティ取得までに待つ最大の待ち時間を表します。

並列利用

ファシリティの利用を並列的に行うか逐次的に行うかを表します。

- ”逐次”は、ファシリティリストに対して 1 つだけリクエストを出します。リクエストが成立するまで、新しいリクエストは出しません。
- ”並列”は、ファシリティリストに対して並列に（複数の）リクエストを出します。Inf を指定すると並列数の上限なくリクエストを出します。

ファシリティ利用の記録

ファシリティの利用の可否とファシリティを利用するまでに要した時間の記録の仕方を表します。

- ”なし”は、記録しません。
- ”モニター”は、ファシリティの利用の可否とファシリティを利用するまでに要した時間を記録します。
- ”時系列モニター”は、ファシリティの利用の可否とファシリティを利用するまでに要した時間を時系列的に記録します。

6.9 ストアから取得 (GetFromStore)



図 6.11: ストアから取得

部品「ストアから取得」(図 6.11) はストア (8.4 章) から指定した個数だけ離散量資源を取得します。部品「ストアへ追加」(6.10 章) と対になる部品です。

ストアリスト

取得するストアを表します。

待ち受け方法

ストアの待ち受け方法を表します。

- "AND" は、順番に関わらず全てのストアから取得を行いません。
- "OR" は、順番に関わらずいずれかのストアから取得を行います。
- "SEQUENCE" は、指定したストアから指定した順に取得を行いません。

取得個数

ストアから取得する個数を表します。

優先度

ストアから取得する際の優先度を表します。

最大待ち時間

取得するまでに許す最大の待ち時間を表します。取得にそれ以上の時間が掛かる場合、取得しません。タイムアウトが発生した時は、タイムアウト時出力ポートに出力されます。

取得した離散量資源

取得した離散量資源をアイテム属性として取得するか、またはアイテムとして出力するかどうかを表します。

- "属性名" は、取得した離散量資源を指定した属性に設定します。
- "出力ポート" は、取得した離散量資源を指定した出力ポートに出力します。ただし、タイムアウトが発生した時は、タイムアウト時出力ポートに空のアイテムリストが出力されます。この場合、この部品に入力されたアイテムは破棄されます。

並列処理

ストアからの取得を並列的に行うか逐次的に行うかを表します。

- ”並列”は、ストアから並列的に離散量資源を取得します。
- ”逐次”は、ストアから逐次的に離散量資源を取得します。取得できるまで、次の取得を行いません。

取得の記録

取得の可否、取得に要した時間、取得個数の記録の仕方を表します。

- ”なし”は、記録しません。
- ”モニター”は、取得の可否、取得に要した時間、取得個数を記録します。
- ”時系列モニター”は、取得の可否、取得に要した時間、取得個数を時系列的に記録します。

6.10 ストアへ追加 (PutToStore)



図 6.12: ストアへ追加

部品「ストアへ追加」(図 6.12) はストア (8.4 章) へ離散量資源を格納します。部品「ストアから取得」(6.9 章) と対になる部品です。

ストアリスト

格納するストアを表します。

待ち受け方法

ストアの待ち受け方法を表します。

- ”AND”は、順番に関わらず全てのストアに格納を行ないます。
- ”OR”は、順番に関わらずいずれかのストアに格納を行ないます。
- ”SEQUENCE”は、指定したストアに指定した順に格納を行ないます。

追加するアイテム

格納する離散量資源を表します。

- "アイテムを追加する"は、アイテム自身を離散量資源として追加します。この場合、空のアイテムが出力されます。
- "アイテム属性を追加する"は、アイテム属性を離散量資源として追加します。この場合、アイテム属性を None にしたアイテムが出力されます。

追加個数

ストアへ追加する個数を表します。

優先度

ストアへ追加する際の優先度を表します。

最大待ち時間

追加するまでに許す最大の待ち時間を表します。追加にそれ以上の時間が掛かる場合、取得しません。

並列処理

ストアへの格納を並列的に行うか逐次的に行うかを表します。

- "並列"は、ストアへ並列的に離散量資源を格納します。
- "逐次"は、ストアへ逐次的に離散量資源を格納します。格納できるまで、次の格納を行いません。

追加の記録

追加の可否、追加に要した時間、追加個数の記録の仕方を表します。

- "なし"は、記録しません。
- "モニター"は、追加の可否、追加に要した時間、追加個数を記録します。
- "時系列モニター"は、追加の可否、追加に要した時間、追加個数を時系列的に記録します。



図 6.13: タンクから取得

6.11 タンクから取得 (GetFromTank)

部品「タンクから取得」(図 6.13) はタンク (8.5 章) から指定した量だけ連続量資源を取得します。部品「タンクへ追加」(6.12 章) と対になる部品です。

タンクリスト

取得するタンクを表します。

待ち受け方法

タンクの待ち受け方法を表します。

- "AND" は、順番に関わらず全てのタンクから取得を行いません。
- "OR" は、順番に関わらずいずれかのタンクから取得を行いません。
- "SEQUENCE" は、指定したタンクから指定した順に取得を行いません。

取得量

タンクから取得する量を表します。

優先度

タンクから取得する際の優先度を表します。

最大待ち時間

取得するまでに許す最大の待ち時間を表します。取得にそれ以上の時間が掛かる場合、取得しません。

取得した連続量資源

取得した連続量資源を属性に持たせるかどうかを表します。指定した属性に取得した連続量資源を加算します。

並列処理

タンクからの取得を並列的に行うか逐次的に行うかを表します。

- ”並列”は、タンクから並列的に連続量資源を取得します。
- ”逐次”は、タンクから逐次的に連続量資源を取得します。取得できるまで、次の取得を行いません。

取得の記録

取得の可否、取得に要した時間、取得量の記録の仕方を表します。

- ”なし”は、記録しません。
- ”モニター”は、取得の可否、取得に要した時間、取得量を記録します。
- ”時系列モニター”は、取得の可否、取得に要した時間、取得量を時系列的に記録します。

6.12 タンクへ追加 (PutToTank)



図 6.14: タンクへ追加

部品「タンクへ追加」(図 6.14) はタンク (8.5 章) へ連続量資源を追加します。部品「タンクから取得」(6.11 章) と対になる部品です。

タンクリスト

追加するタンクを表します。

待ち受け方法

タンクの待ち受け方法を表します。

- ”AND”は、順番に関わらず全てのタンクに追加します。
- ”OR”は、順番に関わらずいずれかのタンクに追加します。
- ”SEQUENCE”は、指定したタンクに指定した順に追加します。

追加量

タンクへ追加する量を表します。

優先度

タンクへ追加する際の優先度を表します。

最大待ち時間

追加するまでに許す最大の待ち時間を表します。追加にそれ以上の時間が掛かる場合、取得しません。

追加する連続量資源

追加した連続量資源をアイテム属性から差し引くかどうかを表します。指定した属性から追加した連続量資源を差し引きます。

並列処理

タンクへの追加を並列的に行うか逐次的に行うかを表します。

- ”逐次”は、タンクへ逐次的に連続量資源を追加します。追加できるまで、次の追加を行いません。
- ”並列”は、タンクへ並列的に連続量資源を追加します。Infを指定すると並列数の上限なく追加を行います。

追加の記録

追加の可否、追加に要した時間、追加量の記録の仕方を表します。

- ”なし”は、記録しません。
- ”モニター”は、追加の可否、追加に要した時間、追加量を記録します。
- ”時系列モニター”は、追加の可否、追加に要した時間、追加量を時系列的に記録します。



図 6.15: 結合

6.13 結合 (Batch)

部品「結合」(図 6.15) は、入力されたアイテムを指定された結合方法と結合条件に従ってフローアイテムリストとして結合します。作成されたフローアイテムリストはアイテムの入力順を保持しており、部品「分割」(6.15 章)で分割可能です。結合の記録として、実際に結合したアイテム数と結合に要した時間を記録できます。

この部品では、まず入力ポートからアイテムを受信し結合条件毎にふるい分けします。ふるい分けられると同時に、並列して結合方法に従ってフローアイテムリストとして結合を行います。結合方法の”結合数”、”結合間隔”、またはそのいずれかが成立した段階で、それまでに結合したフローアイテムリストを出力ポートに出力します。記録を指定した場合、出力するフローアイテムリストの長さ、フローアイテムリストの先頭のアイテムが入力されてからフローアイテムリストとして出力するまでの時間を記録します。

「結合」は、フローアイテム「アイテム」(9.2 章)や部品「マージ」(6.14 章)に関連します。

結合方法

アイテムの結合の仕方を表します。受信したアイテムを指定したタイミングで結合します。

- ”結合数”は、「結合条件」が一致するアイテム数が指定個数になるまで結合します。
- ”結合間隔”は、「結語条件」が一致するアイテムで分けた中で 1 つめのアイテムが到着してから経過した時間内に到着したアイテムを結合します。

結合条件

結合するアイテムを表します。条件が成り立つアイテム群で結合します。

- ”なし”は、全てのフローアイテムを区別せずに「結合方法」で結合します。
- ”属性”は、属性の値が等しいアイテムで区別し「結合方法」で結合します。

結合の記録

実際に結合したアイテム数と結合に要した時間の記録の仕方を表します。

- ”なし”は、記録しません。
- ”モニター”は、結合した時点の結合したアイテム数と結合に要した時間を記録します。”結合数”が成立して結合されていれば、指定した”結合数”と結合したアイテム数は一致します。同様に、”結合間隔”が成立して結合されていれば、指定した”結合間隔”と結合に要した時間は一致します。
- ”時系列モニター”は、結合した時点の結合したアイテム数と結合に要した時間を時系列で記録します。”結合数”が成立して結合されていれば、指定した”結合数”と結合したアイテム数は一致します。同様に、”結合間隔”が成立して結合されていれば、指定した”結合間隔”と結合に要した時間は一致します。

6.14 マージ (Merge)



図 6.16: マージ

部品「マージ」(図 6.16) はアイテム属性を要約することでアイテムを結合します。全ての属性に対して要約値が計算されます。

計算方法

アイテム属性の要約値の計算の仕方を表します。

結合方法

アイテムの結合の仕方を表します。

- ”結合数”は、アイテム数で結合します。
- ”結合間隔”は、1 つめのアイテムが到着してから経過した時間内に到着したアイテムを結合します。
- ”結合数 or 結合間隔”は、上記のいずれかの条件が満たされた時点でアイテムを結合します。

結合条件

結合するアイテムを表します。

- ”なし”は、全てのフローアイテムを結合します。
- ”属性”は、属性の値が等しいアイテムを結合します。

結合の記録

結合したアイテム数とその時間の記録の仕方を表します。

- ”なし”は、記録しません。
- ”モニター”は、結合した時点の結合したアイテム数と結合に要した時間を記録します。
- ”時系列モニター”は、結合した時点の結合したアイテム数と結合に要した時間を時系列で記録します。

6.15 分割 (Split)



図 6.17: 分割

部品「分割」(図 6.17) はフローアイテムリストを分割します。「結合」(6.13 章) や部品「生成」(6.2 章) でフローアイテムリストが作成されます。

分割方法

再帰的に分割を行うかどうかを表します。

6.16 設定 (Assign)



図 6.18: 設定

部品「設定」(図 6.18) は、アイテムの属性やパラメータに値を設定します。入力されたアイテムの既存のアイテム属性やシミュレーションモデル固有の

パラメータを更新できます。また、入力されたアイテムに新しくアイテム属性を割付けたり、シミュレーションモデル固有のパラメータを新しく割り付けたりできます。アイテム属性とパラメータの設定を同時に行えます。その際、アイテム属性の設定の後にパラメータの設定を行うため、互いに値を参照しあう場合は注意する必要があります。アイテムがフローアイテムリストであった場合、そのリスト内の全てのアイテムが対象になります。

この部品内特有の時間経過はなく、入力ポートからアイテムを受信した順に設定を行い、出力ポートに出力します。パラメータのみに値の設定を行う場合でも、設定を行うタイミングは入力ポートからアイテムを受信したときになります。

「設定」は、フローアイテム「アイテム」(9.2章)やパラメータ(5.4章)に関連します。

アイテム属性

「名前」が値の更新または新しく割付けするアイテム属性を表します。既存の属性名であれば値の更新、新しい属性名であれば値の割付になります。複数のアイテム属性を同時に設定できます。このリスト順にアイテム属性が設定されるため、同じアイテム属性が指定された場合は後続の設定が優先されます。フローアイテムリストの場合、リスト内の全てのアイテムのアイテム属性が対象になります。

なお、特殊なアイテム属性名として、`_item_color` と `_item_marker` があります。これらはアニメーション表示モードの時のみ有効で、`_item_color` はアイテム表示する際の色、`_item_marker` はアイテム表示する際のマーカーを指定します。

パラメータ

値の更新または新しく割付けするパラメータを表します。既存のパラメータであれば値の更新、新しいパラメータであれば値の割付になります。複数のパラメータを同時に設定できます。このリスト順にパラメータが設定されるため、同じパラメータが指定された場合は後続の設定が優先されます。

6.17 判断 (Decide)

部品「判断」(図 6.19) は、入力されたアイテムの出力先を条件に従って分岐させます。条件式が成立する場合、その条件式で指定した出力ポートにア



図 6.19: 判断

アイテムを出力します。リストの先頭ほど優先順位が高く、複数の条件式が成立した場合、より優先順位が高い出力ポートにアイテムを出力します。

この部品内特有の時間経過はなく、入力ポートからアイテムを受信した順に条件式が成立するかを評価します。指定した条件式リスト順に条件式の成立を評価し、成立した条件式があれば、アイテムをその条件式に紐付けた出力ポートに即座に出力します。以降の条件式リストは評価しません。条件式が何らかの理由で評価できなかった場合、その条件式は成立しなかったものとみなします。

条件式リスト

条件式と出力ポートの組のリストを表します。指定した条件式が成立した場合に、対応する出力ポートにアイテムが出力されます。条件式リスト順に優先順位が設定されており、複数の条件式が成立する場合、条件式リストのより先頭側の条件式が成立したものとみなされ、対応する出力ポートにアイテムを出力します。

それ以外

条件式リストの条件式が全て成り立たなかった場合に、アイテムを出力する出力ポートを表します。

6.18 遅延 (Delay)



図 6.20: 遅延

部品「遅延」(図 6.20) は、入力されたアイテムの出力を遅延させます。アイテムを出力するのを指定時間だけ遅延させたり、アイテムを出力した後に次のアイテムを受信するのを指定時間だけ遅延させたりできます。また、個々のアイテムを並列的に遅延させるか逐次的に遅延させるかを選択できます。並列的にアイテムの出力を遅延させる場合、指定時間だけ個々のアイテムの出力は遅延します。この場合、「出力後の待ち時間」は無効になります。逐次

的にアイテムの出力を遅延させる場合、指定時間に加えてアイテムが処理されるまでの待ち時間だけ個々のアイテムの出力は遅延します。

入力ポートからアイテムを受信して「出力前の待ち時間」だけ待ち受けし、待ち受けの成立後に出力ポートにアイテムを出力して、さらに「出力後の待ち時間」だけ待ち受けを行います。並列的にアイテムの出力を遅延させる場合、受信したアイテム毎にこのサイクルをそれぞれ独立したプロセスで実行します。逐次的にアイテムの出力を遅延させる場合、このサイクルを順に行います。サイクルを実行中にアイテムが入力ポートに入力されていても、このサイクルが完了するまで次のサイクルは実行しません。入力キューのサイズが 0 の場合、この入力ポートに入力しようとするプロセスの実行もこのサイクルが終わるまで待たされるため、注意が必要です。

出力前の待ち時間

アイテムを出力する前の最短待ち時間を表します。アイテムを受信してから、この指定時間だけ出力を遅延させます。逐次的の場合、入力キューでの待ち時間だけ遅延が生じている場合があります。

出力後の待ち時間

アイテムを出力した後の最短待ち時間を表します。アイテムを出力してから、この指定時間だけ次のアイテムの受信を遅延させます。並列の場合、この指定時間は無効になります。

並列処理

「並列処理」は、アイテム出力の遅延を並列的に行うか逐次的に行うかを表します。

- ”並列”は、アイテムの出力を並列的に遅延させます。アイテムは並列的に遅延されて出力されるため、「出力後の待ち時間」は無効になります。
- ”逐次”は、アイテムの出力を逐次的に遅延させます。アイテムの遅延が完了するまで、次のアイテムの遅延を行いません。そのため指定した時間以上に遅延が生じる場合があります。



図 6.21: 複製

6.19 複製 (Duplicate)

部品「複製」(図 6.21) は、入力されたアイテムを指定した個数だけ指定した方法で複製します。複製方法は、アイテムの参照を複製する方法、アイテムの構造を複製する方法、アイテムを複製する方法の 3 つの方法があります。アイテムの参照を複製すると、複製対象のアイテムを繰り返し出力できます。出力先を変えれば、複製対象のアイテムを異なる複数のフローに同時に流すことができます。アイテムの構造を複製すると、複製対象のアイテムと同一の構造を持ったアイテムを繰り返し出力できます。複製対象のアイテムの属性は複製しないため、出力したアイテム間で属性値は同一になります。属性に設定されているオブジェクトの状態を変更する場合には、注意が必要です。アイテムを複製すると、複製対象のアイテムの構造と内容を複製したアイテムを繰り返し出力できます。またアイテムを複製した際に、全ての出力ポートでの出力が完了してから次のアイテムを複製するのか、完了を待たずに次のアイテムを複製するのかを選択できます。

この部品内特有の時間経過はなく、入力ポートからアイテムを受信した順に指定された複製方法で、全ての出力ポートに指定された複製数だけ複製し出力します。出力はサブプロセスで行い、全ての出力ポートでの出力が完了してからの場合は完了するまで待ち受けし、完了を待たない場合はサブプロセスを並列に実行し、次のアイテムの複製に移ります。

複製数

出力ポート毎に複製する数を表します。負の場合、複製数は 0 と見なします。

複製方法

アイテムの複製方法を表します。

- "参照" は、アイテムの参照を複製します。アイテムは複製されず、複製対象のアイテムを出力します。
- "浅いコピー" は、アイテムの構造を複製します。アイテムの属性は複製されず、複製対象のアイテムの属性と同じ値を設定し出力します。
- "深いコピー" は、アイテムの構造と内容を複製します。アイテムを複製し出力します。

複製タイミング

複製するタイミングを表します。

- ”出力完了後”は、複製したアイテムが全ての出力ポートで出力完了してから次のアイテムを複製します。
- ”逐次”は、複製したアイテムが全ての出力ポートで出力完了していなくても次のアイテムを複製します。

6.20 通過 (Pass)



図 6.22: 通過

部品「通過」(図 6.22) はアイテムを無条件で通過させます。

6.21 記録 (Record)



図 6.23: 記録

部品「記録」(図 6.23) はアイテムの個数や累積個数、アイテム属性を含む式の値、アイテムの到着時間間隔を記録します。

モニター種類

どのモニターを用いるかを表します。

- ”モニター”は、モニターを用いて記録します。
- ”時系列モニター”は、時系列モニターを用いて記録します。

記録内容

記録対象を表します。

- ”列名”は、記録内容の名前を表します。
- ”列の型”は、記録内容の型を表します。
- ”記録方法”は、アイテムの何を記録するかを表します。

- カウントは、この部品に 1 度に入力されたアイテムの個数を記録します。
- 累積カウントは、1 度に入力されたアイテムの個数を累積的に記録します。
- 式は、アイテム属性を含む任意の式の値を記録します。
- 累積式は、アイテム属性を含む任意の式の値を累積的に記録します。
- 時間間隔は、アイテムの到着時間間隔を記録します。

6.22 連続 (Continuous)



図 6.24: 連続

部品「連続」(図 6.24) は、連続シミュレーションを行います。

連続シミュレーションとは、常微分方程式 (ODE, Ordinary Differential Equation) の初期値問題 (IVP, Initial Value Problem) の数値解法に相当し、以下のような、時間 t_0 で、 y が y_0 であった場合に、 $y = y(t)$ ($t_0 \leq t \leq T$) の近似解を求める問題を解きます。

$$\begin{aligned}\frac{dy}{dt} &= f(t, y) \quad (t_0 \leq t \leq T) \\ y(t_0) &= y_0\end{aligned}$$

k 階の常微分方程式は以下のように表されます。

$$\begin{aligned}y^{(k)} &= f(t, y, y', y'', \dots, y^{(k-1)}) \\ y(t_0) &= y_0 \\ y'(t_0) &= y'_0 \\ y''(t_0) &= y''_0 \\ &\vdots \\ y^{(k-1)}(t_0) &= y_0^{(k-1)}\end{aligned}$$

ここで、 $y = y^0, y' = y^1, \dots, y^{(k-1)} = y^{k-1}$ と置けば、次のような 1 階常微分方程式に変換する事が出来ます。

$$\begin{aligned}
 (y^{k-1})' &= f(t, y^0, y^1, y^2, \dots, y^{k-1}) \\
 (y^{k-2})' &= y^{k-1} \\
 &\vdots \\
 (y^1)' &= y^2 \\
 (y^0)' &= y^1
 \end{aligned}$$

ローカル変数

この部品内でローカルに使われる連続変数がある場合に指定します。

- "変数名" はローカル変数の名前を表します。
- "初期値" はローカル変数の初期値を表します。
- もし、シミュレーション終了時に、そのローカル変数の値を記録しておきたい場合は、"終了時の更新" に指定します。
- "最大値" はローカル変数の最大値を表します。
- "最小値" はローカル変数の最小値を表します。

微分方程式

ここに指定した微分方程式に従い連続シミュレーションを行います。

- "参照変数" は微分方程式内で参照したい連続変数を表します。
- "式" には連立微分方程式を指定します。各式の左辺にはどの連続変数の導関数を定義するかを設定します。右辺には、時間を示す T、左辺に指定した連続変数、"参照変数" に指定した連続変数を使った式を指定します。左辺にローカル変数を指定した場合、ローカル変数 [i] のように、インデックス付きの変数が表示されます。右辺にて、"ローカル変数 [i]" と記述すれば、ローカル変数値を示します。添字無しで"ローカル変数" と記述した場合、現時点で有効な全てのローカル変数のリストを示します。"ローカル変数.items" と記述した場合、現時点で有効な全てのローカル変数が作成された時のフローアイテムのリストを示します。

出力先

6.23 即時報酬確定 (GetReward)

連続シミュレーションの終了条件とその出力先を指定します。

- ”終了条件が成立”の場合は”参照変数”を使った”条件式”が成立したら、指定した”出力先”に出力します。
- ”下限”の場合は”変数”が下限に達して連続シミュレーションが終了したら、指定した”出力先”に出力します。
- ”上限”の場合は”変数”が上限に達して連続シミュレーションが終了したら、指定した”出力先”に出力します。

それ以外の出力先

どの条件にも適合しなかった場合は、ここに指定した出力先に出力します。

6.23 即時報酬確定 (GetReward)



図 6.25: 即時報酬確定

部品「即時報酬確定」(図 6.25) では、強化学習を用いるシミュレーションにおいて、入力されたアイテムに対応する行動の即時報酬値を決定します。

本部品では、アイテム属性に設定された報酬決定関数を呼び出して即時報酬値を決めます。報酬決定関数は、強化学習モデルの利用者(強化学習セレクト (7.2.7 章) や部品「強化学習設定」(6.24 章)) で、即時報酬値の確定を行うタイミングを次のアイテムの行動決定時点からずらす場合にアイテムの属性に設定されています。タイミングをずらさない際には本部品は不要です。

強化学習の学習サイクルの都合上、一つの行動に対して丁度一回だけ即時報酬値を求めて評価する必要があります。例えば、アイテムが循環するシミュレーションにおいては、行動と即時報酬を交互に求めるようなモデルを組み、行動を再設定していないアイテムが本部品に続けて入力されることの無いようにします。

報酬決定関数のアイテム属性名

報酬決定関数が割付けられたアイテム属性を指定します。指定された属性に報酬決定関数が設定されていなければ、警告を出力します。フローアイテムリストの場合、リスト内の全てのアイテムのアイテム属性が対象になります。

6.24 強化学習設定 (AssignRL)



図 6.26: 強化学習設定

部品「強化学習設定」(図 6.26) は、入力されたアイテムの属性に、強化学習モデルが選んだ行動値を設定します。

強化学習モデル

利用する強化学習モデルを指定します。シミュレーションモデルに設定された離散イベント用強化学習モデルが選択できます。

エピソード

強化学習におけるエピソードの区別に関する設定を表します。

- ”区別しない”は、シミュレーション全体を一つのエピソードとみなして学習を行い、エピソード名には本アイコンの名前が割り当てられます。
- ”区別する”は、文字列を設定してエピソードを定めます。この際、入出力対象であるフローアイテムを item として参照できます。例えば、アイテムのある属性 foo でエピソードを区別したい場合は、item.foo と設定します。区別する際の文字列として、デフォルトでは item が指定されます。

行動値を設定するアイテム属性名

強化学習モデルを利用して求めた行動値を設定するアイテム属性の名前を指定します。フローアイテムリストの場合、リスト内の全てのアイテムのアイテム属性が対象になります。

即時報酬計算のタイミングをずらすか

アイテムに設定した行動値に対して、強化学習モデルが即時報酬値を確定するタイミングを、次のアイテムの行動値を設定する時点からずらすかどうか指定します。

- ”ずらさない”は、即時報酬値を確定するタイミングを、次のアイテムの行動値を設定する時点とします。

- ”ずらす”は、即時報酬値を確定するタイミングを、部品「即時報酬確定」(6.23章)にアイテムが到着した時点とします。具体的には、呼び出すと報酬が決定される関数(報酬決定関数)をアイテムの属性に設定し、即時報酬確定部品に到着した時に報酬決定関数が呼び出されます。
 - － ”報酬決定関数を設定するアイテム属性名”で指定するアイテム属性に報酬決定関数を設定します。アイテムがフローアイテムリストであった場合、そのリスト内の全てのアイテムが対象になります。

第7章 リンク

7.1 ポートの動作

入力ポートは部品を構成する要素のひとつで、他の出力ポートからリンクが結ばれていればフローアイテムを入力します。キューを持つ場合と持たない場合があります。キューを持たない場合はセレクトタによってどの出力ポートから入力するかを決定します。部品は複数の入力ポートを持ちますがひとつも存在しない場合もあります。

出力ポートは部品を構成する要素のひとつで、他の入力ポートにリンクが結ばれていればフローアイテムを出力します。キューを持つ場合と持たない場合があります。キューを持たない場合はセレクトタによってどの入力ポートへ出力するかを決定します。部品は複数の出力ポートを持ちますがひとつも存在しない場合もあります。

部品共通設定タブのポートのキューサイズが0でないなら、そのポートはキューを持ちます。0の場合は、そのポートはキューを持ちません。入力ポートのキューは必ず FIFO キュー (20 章) になります。

7.2 セレクトタ

入力ポートのセレクトタの場合、その入力ポートに入力する複数のリンク $\{L_i\}$ ($1 \leq i \leq n$) から、セレクトタはひとつの L_k を選択し、そのリンクの先の出力ポートからフローアイテムを入力します。

出力ポートのセレクトタの場合、その出力ポートから出力する複数のリンク $\{L_i\}$ ($1 \leq i \leq n$) から、セレクトタはひとつの L_k を選択し、そのリンクの先の入力ポートへフローアイテムを出力します。

セレクトタとしては、以下が利用可能で、それぞれ部品共通設定タブのポートの選択方式で指定出来ます。

7.2.1 ランダム (一様)

確率 $P(L_i) = 1/n$ でリンク L_i を選びます。

7.2.2 ランダム (重み付き)

重みを $\{w_i\}$ とすると、確率 $P(L_i) = w_i / \sum w_i$ でリンク L_i を選びます。

7.2.3 ラウンドロビン

セレクタが呼ばれた回数を k (≥ 1) とすると k 番目に リンク $L_j, j = ((k-1) \bmod n) + 1$ を選びます。

7.2.4 固定

常に指定されたリンクを選びます。

7.2.5 最短キュー

リンクの先のポートのキューが最も少ないリンクを選びます。

ただし、最小キューサイズのリンクが複数ある場合は、セレクタが呼ばれた回数を k (≥ 1) とし、最小キューサイズのリンクのインデックス集合を $\{a_i\}$ とすると、リンク $L_{a_m}, m = \arg \min_i ((k - a_i + 1) \bmod n)$ を選択します。

7.2.6 ユーザ定義

ユーザが定義したセレクタを呼び出します。

ユーザ定義セレクタは以下のように書く事が出来ます。

```
def userSelector():
    # 必要なら前処理
    # ユーザ定義の変数定義
    def select(port, item = False):
        # セレクターは入力ポートで使われる場合と出力ポートで
        # 使われる場合がある。
        # (出力ポートの場合は item にフローアイテムが入っている)
        # port.links でリンクリストが取得出来る
        # それを使って、
        # * 入力ポートなら入力するリンクをひとつ返す
        # * 出力ポートなら出力するリンクをひとつ返す
        # あるいは False を返す事も出来るが、その場合、
        # 永久にその入/出力ポートからは入/出力しない。
    return select
```

例えば、あるリンクを一様に選ぶセクタは以下のように定義する事が出来ます。

```
def userSelector():
    r = uniformDistribution()
    def select(port, item = False):
        links = port.links
        if len(links) <= 0:
            return False
        else:
            index = int(next(r) * len(links))
            return links[index]
    return select
```

7.2.7 強化学習

強化学習によりリンクを選択します。強化学習によるセクタを使用する場合は、以下を設定する必要があります。

強化学習モデル

使用する強化学習モデルを指定します。シミュレーションモデルに設定された離散イベント用強化学習モデルが選択できます。

エピソード

強化学習におけるエピソードの区別に関する設定を行います。区別しない場合は、シミュレーション全体を1つのエピソードとみなして学習が行われます。区別する場合は、エピソードを区別する文字列を設定します。文字列の指定の際には、入出力対象のフローアイテムを `item` として参照できます。例えば、フローアイテムの属性 `id` でエピソードを区別したい場合は、`item.id` と設定します。

行動とリンクの対応

強化学習モデルにおける行動とリンクとの対応付けを設定します。

即時報酬計算のタイミングをずらすか

強化学習が選択した、フローアイテムの行動に対する即時報酬値の計算を、次のフローアイテムの行動の選択時点からずらすかどうかを設定します。ずらす際には、どの名前のアイテム属性に報酬決定関数を設定するか、“報酬決定関数を設定するアイテム属性名”で設定します。この報酬決定関数を呼び出すタイミングで即時報酬値の計算が行われます。

7.3 接続方式

部品間をリンクで結びシミュレーションモデルを作成しますが(5章)、リンクの結び方によって動作が変わってきます。リンクの結び方を接続方式と呼びます。ここでは接続方式のパターンとその動作を説明します。

接続方式には以下のようなものがあります。

7.3.1 一対多 (出力キュー 0、入力キュー m)

図 7.1 において、 $m_i > 0, 1 \leq i \leq k, k > 0$ の場合、

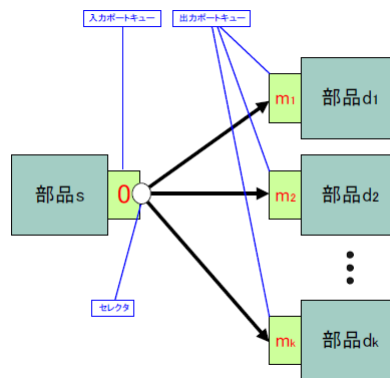


図 7.1: 接続方式 1

フローアイテムを部品 s の出力ポートへ出力すると、セレクタによって部品 d_i の入力ポートの何れかにフローアイテムを出力します。もし、部品 d_i の入力ポートのキューが一杯なら、出力出来ない状態でブロックします。

部品 d_i の入力ポートから入力を行うと、部品 d_i の入力ポートのキューが空でなければ入力し、空の場合は入力出来ない状態でブロックします。

最も良く使われる接続方式です。例えば、銀行において窓口が複数あり、お客は最も空いている窓口に並ぶ場合などがあります。

7.3.2 一対多 (出力キュー 0、入力キュー 0)

図 7.2 において、 $k > 0$ の場合、

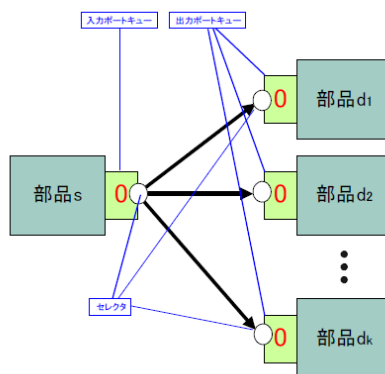


図 7.2: 接続方式 2

フローアイテムを部品 s の出力ポートへ出力すると、セレクタによって部品 d_i の入力ポートの何れかにフローアイテムを出力しようとしてします。もし、部品 d_i の入力ポートが受信要求をしていたら、フローアイテムが流れますが、部品 d_i の入力ポートが受信要求をしていない場合は、出力出来ない状態でブロックします。

うまく使わないと部品 s にフローアイテムがたまってしまうので注意が必要です。 $m = 1$ の場合はよく使われます。例えば、物流システムにおいて、配送工程 1 と配送工程 2 の間などがあります。

7.3.3 一対多 (出力キュー n 、入力キュー m)

図 7.3 において、 $n > 0, m_i > 0, 1 \leq i \leq k, k > 0$ の場合は、部品 s の出力ポートのキューに全てのフローアイテムがたまってしまい、フローアイテムが流れません。更にキューが一杯になると部品 s の入力も滞ってしまいます。通常は誤った接続方式です。

7.3.4 一対多 (出力キュー n 、入力キュー 0)

図 7.4 において、 $n > 0, k > 0$ の場合、

フローアイテムを部品 s の出力ポートへ出力すると、部品 s の出力ポートのキューにフローアイテムをたくわえます。部品 s の出力ポートが一杯なら出力出来ずにブロックします。

7.3 接続方式

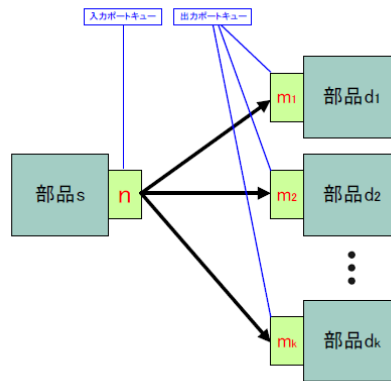


図 7.3: 接続方式 3

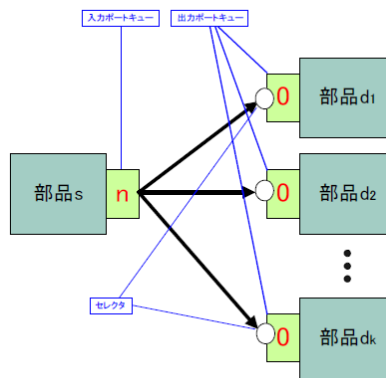


図 7.4: 接続方式 4

部品 d_i の入力ポートから入力を行うと、部品 i の入力ポートのセレクタによって入力先を選びますが、この場合は常に部品 s の出力ポートのキューから入力します。

よく使われる接続方式です。例えば、コールセンターにおいてオペレーターが自分で電話を取る場合などがあります。

7.3.5 多対一 (出力キュー 0、入力キュー m)

図 7.5 において、 $j > 0, m > 0$ の場合、

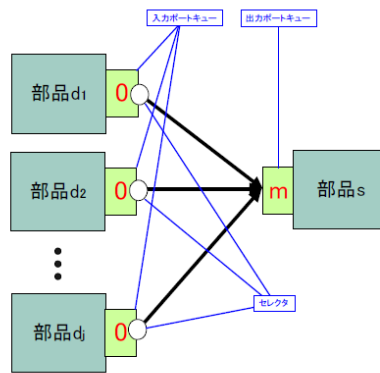


図 7.5: 接続方式 5

フローアイテムを部品 d_i の出力ポートへ出力すると、セレクタによって出力先を選びますが、この場合は常に部品 s へ出力します。部品 s の入力ポートのキューが一杯なら出力出来ない状態でブロックします。

部品 s の入力ポートから入力を行うと、部品 s の入力ポートのキューが空でなければ入力し、空の場合は入力出来ない状態でブロックします。

よく使われる接続方式です。例えば、オフィスにおいて、ひとつのプリンタを複数の社員が待ち行列を作って使用する場合があります。

7.3.6 多対一 (出力キュー 0、入力キュー 0)

図 7.6 において、 $j > 0$ の場合、

フローアイテムを部品 d_i の出力ポートへ出力すると、セレクタによって出力先を選びますが、この場合は常に部品 s へ出力しようとします。もし、部品 s の入力ポートが受信要求をしていたら、フローアイテムが流れますが、部品 s の入力ポートが受信要求をしていない場合は、出力出来ない状態でブロックします。

7.3 接続方式

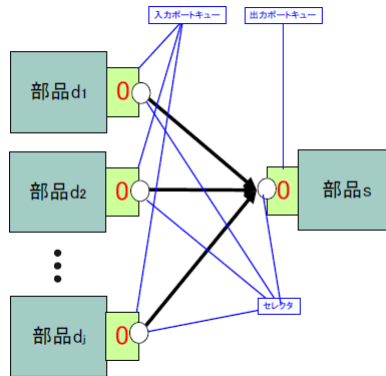


図 7.6: 接続方式 6

うまく使わないと部品 d_i にフローアイテムがたまってしまいますので注意が必要です。 $m = 1$ の場合はよく使われます。例えば、製品開発ラインにおいて、工程 1 と工程 2 の間で、その間にバッファ(キュー)がない場合などがあります。

7.3.7 多対一 (出力キュー n 、入力キュー m)

図 7.7 において、 $m > 0, n_i > 0, 1 \leq i \leq j, j > 0$ の場合は、部品 d_i の出力ポートのキューに全てのフローアイテムがたまってしまい、フローアイテムが流れません。更にキューが一杯になると部品 d_i の入力も滞ってしまいます。通常は誤った接続方式です。

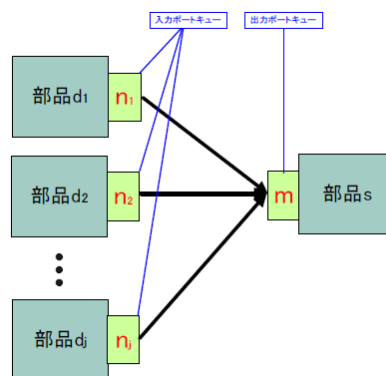


図 7.7: 接続方式 7

7.3.8 多対一 (出力キュー n 、入力キュー 0)

図 7.8 において、 $n_i > 0, 1 \leq i \leq j > 0$ の場合、

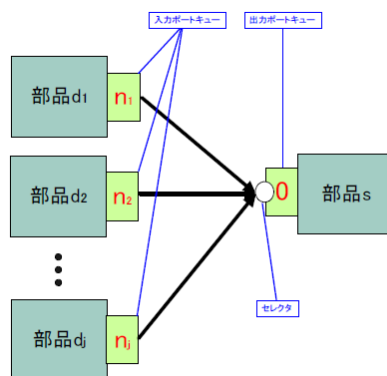


図 7.8: 接続方式 8

フローアイテムを部品 d_i の出力ポートへ出力すると、部品 d_i の出力ポートのキューにフローアイテムをたくわえます。部品 d_i の出力ポートが一杯なら出力出来ずにブロックします。

部品 s の入力ポートから入力を行うと、セレクトラによって部品 d_i の出力ポートの何れかからフローアイテムを入力します。部品 s の入力ポートのキューが空なら入力出来ない状態でブロックします。

うまく使わないと部品 d_i にフローアイテムがたまってしまいますので注意が必要です。例えば、ルータにおいて、複数の入力をラウンドロビンで処理する場合などがあります。

7.3.9 多対多

場合によっては多対多の接続もあります。例えば、一对多 (出力キュー 0 、入力キュー m) (7.3.1 章) などは出力側を多にする事で多対多の接続にも拡張出来ます。しかし、ひとつでも入/出力ポート双方にキューがあるようなリンクが存在する場合、フローアイテムがたまってしまうため、通常は誤った接続です。

または、多対一と一对多をつなげて多対多を構成する場合があります。その場合は中央の部品 (例えば「通過」部品) が中継地点になります。中央の部品の入力ポートあるいは出力ポートにキューがあれば、多対多の中継キューになります。なお、このような接続は、ストアの入出力部品 (ストアへ追加、ストアから取得) を使っても表現する事が出来ます。また、中央の部品にキューがない場合は、 $n \cdot m$ 本のリンク貼るのと同じ事を表現します。

第8章 資源

モデル上に配置する資源についてを説明します。

資源とは、「利用するために待ち行列が発生するようなもの」をモデル化したものです。以下のファシリティ、ストア、タンク、イベントがあります。

8.1 資源の編集

資源の下のラベルの以外の場所を左ダブルクリック、あるいは、資源の配置されている場所を右クリックして表示されるメニュー (3.5.1 章) から「編集」を選ぶと、資源編集画面が表示されます。

共通設定タブと属性設定タブで構成されます。

8.1.1 共通設定タブ

共通設定タブでは、図 8.1 のような画面構成になります。

クラス名

自動生成コード (5.8.1 章) におけるこの資源のクラス名になります。編集する事が出来ません。

オブジェクト名

自動生成コード (5.8.1 章) におけるこの資源のインスタンス名になります。通常は編集する必要はありません。

説明

モデル編集画面におけるこの資源の名前になります。5.3.15 章の方法でも変更する事が出来ます。

アニメーション表示条件、アニメーション表示範囲

アニメーションに関するパラメータです。詳細は (26 章) アニメーションにあります。

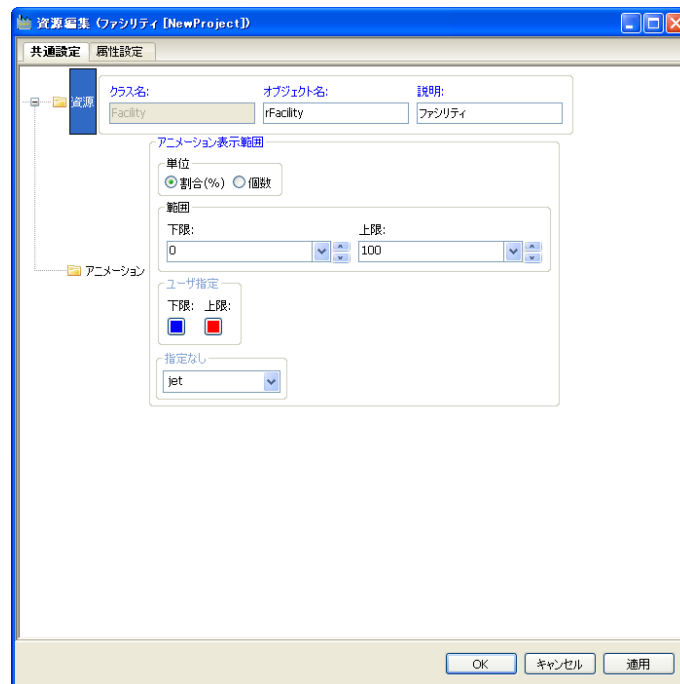


図 8.1: 資源共通設定タブ

8.1.2 属性設定タブ

属性設定タブでは、図 8.2 のような画面構成になります。

上のヘルプボタンを押すと、その資源の説明が表示されます。

資源ごとに表示される属性は変わります。資源の振る舞いを定めます。

8.2 イベント (Event)

イベント (図 8.3) は、発火されるまで待ち受けする同期資源を表します。発火を待つ時に待ち行列が発生します。発火側には待ち行列が発生しません。

例えば、「電話がかかってくる」などです。電話がかかってくるまで電話を待つが、複数の人が電話を待っている場合、並びが速いもの順に受ける場合もあり、優先度が高い順に受ける場合などがあります。

イベントに関する部品としては、部品「イベント待ち」(6.4 章)、部品「イベント発火」(6.5 章) などがあります。

キュー

発火を待つ際に並ぶキューを表します。

記録

8.3 ファシリティ (Facility)

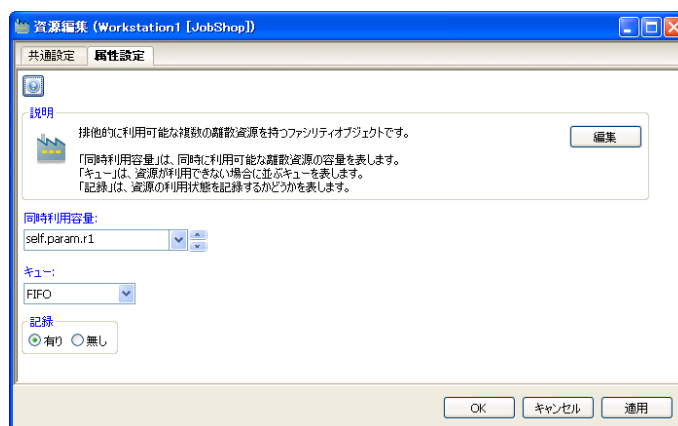


図 8.2: 資源属性設定タブ



図 8.3: イベント

発火待ちの状態を記録するかどうかを表します。

8.3 ファシリティ (Facility)



図 8.4: ファシリティ

ファシリティ(図 8.4) は、排他的に利用可能な複数の離散資源を持つものを表します。資源取得には待ち行列が発生します。資源開放には待ち行列が発生しません。

例えば、「銀行の窓口」などです。銀行の窓口がひとつしかないとなると、ある人がその窓口を占有していると、他の人はその窓口が空くのを待たなければなりません。窓口を占有していた人の処理が終了すればその窓口を開放し、待ち行列に並んでいた次の人が窓口を占有します。

ファシリティに関する部品としては、部品「ファシリティ要求」(6.6 章)、部品「ファシリティ解放」(6.7 章)、部品「ファシリティ利用」(6.8 章)などがあります。

同時利用容量

同時に利用可能な離散資源の容量を表します。

キュー

資源が利用できない場合に並ぶキューを表します。

記録

資源の利用状態を記録するかどうかを表します。

容量変化スケジュール

スケジュール有りを選んだ場合、容量の変化スケジュールを登録する事ができる。開始時オフセット時間は、開始時間を指定する。スケジュールには、表指定か式にてスケジュールを指定する。式の場合、時間と容量からなるタプルのリストを指定する。繰り返しには、繰り返し無し、あるいは、繰り返し時間を指定する。繰り返し時間は、その時間でスケジュールが繰り返される事を意味する。つまり、スケジュールで指定した時間以上を指定しなくてはならない。

停止確率

停止有りを選んだ場合、停止間隔と回復を指定する。停止間隔には、停止間隔を指定する。回復有りを指定した場合、回復時間を指定する。回復無しを選んだ場合は回復しない。

停止時に自動開放

停止時に自動開放を指定した場合、ファシリティが利用不可になった時に、強制的に開放する。

8.4 ストア (Store)



図 8.5: ストア

ストア (図 8.5) は、有限の離散量資源を貯えられるようなものを表します。取得と追加操作が可能で、双方に待ち行列が発生します。取得と追加操作は

非対称で良く、取得しても再追加する必要はなく、追加しても再取得する必要はありません。

例えば、「コンビニエンスストアの棚」などです。コンビニエンスストアの棚には複数の商品がありますが、お店の人は定期的に商品を入荷し配置するし、定期的にお客がその商品を買います。棚が一杯の時は商品を配置出来ず、棚が空の場合は商品を買う事は出来ません。

ストアに関する部品としては、部品「ストアから取得」(6.9章)や部品「ストアへ追加」(6.10章)があります。

容量

離散量資源を貯えられる容量を表します。

入力キュー

離散量資源を貯えるときに並ぶキューを表します。

出力キュー

離散量資源を取り出すときに並ぶキューを表します。

記録

資源の利用状態を記録するかどうかを表します。

容量変化スケジュール

スケジュール有りを選んだ場合、容量の変化スケジュールを登録する事ができる。開始時オフセット時間は、開始時間を指定する。スケジュールには、表指定か式にてスケジュールを指定する。式の場合、時間と容量からなるタプルのリストを指定する。繰り返しには、繰り返し無し、あるいは、繰り返し時間を指定する。繰り返し時間は、その時間でスケジュールが繰返される事を意味する。つまり、スケジュールで指定した時間以上を指定しなくてはならない。

8.5 タンク (Tank)

タンク (図 8.6) は、有限の連続量資源を貯えられるようなものを表します。取得と追加操作が可能で、双方に待ち行列が発生します。取得と追加操作は



図 8.6: タンク

非対称で良く、取得してもそれを再追加する必要はなく、また追加してもそれを再取得する必要はありません。

例えば、「ガソリントank」などです。ある車のガソリントankには、一定量のガソリンが入っているが、車が走ればその量は減り、ガソリンスタンドで給油すれば量が増えます。容量が空の場合は車は走れず、容量が一杯の時はそれ以上給油出来ません。

タンクに関する部品としては、部品「タンクから取得」(6.11 章) や部品「タンクへ追加」(6.12 章) があります。

容量

連続量資源を貯えられる容量を表します。

初期量

始めに貯えている連続量資源の量を表します。

入力キュー

連続量資源を貯えるときに並ぶキューを表します。

出力キュー

連続量資源を取り出すときに並ぶキューを表します。

記録

資源の利用状態を記録するかどうかを表します。

容量変化スケジュール

スケジュール有りを選んだ場合、容量の変化スケジュールを登録する事ができる。開始時オフセット時間は、開始時間を指定する。スケジュールには、表指定か式にてスケジュールを指定する。式の場合、時間と容量からなるタプルのリストを指定する。繰り返しには、繰り返し無し、あるいは、繰り返し時間を指定する。繰り返し時間は、その時間でスケジュールが繰返される事を意味する。つまり、スケジュールで指定した時間以上を指定しなくてはならない。

第9章 フローアイテム

モデル上に配置するフローアイテムについてを説明します。

フローアイテムとは、シミュレーション内に流れるものです。出力ポートからリンクを辿り入力ポートに入り、部品の動作の契機になります。モデル上に配置するのは、フローアイテムの種類 (class) であり、「生成」部品でその実体を作ります。

9.1 フローアイテムの編集

フローアイテムの下のラベルの以外の場所を左ダブルクリック、あるいは、フローアイテムの配置されている場所を右クリックして表示されるメニュー (3.5.1 章) から「編集」を選ぶと、フローアイテム編集画面が表示されます。共通設定タブと属性設定タブで構成されます。

9.1.1 共通設定タブ

共通設定タブでは、図 9.1 のような画面構成になります。

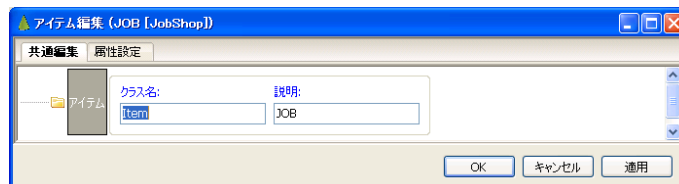


図 9.1: フローアイテム共通設定タブ

クラス名

自動生成コード (5.8.1 章) におけるこのフローアイテムのクラス名になります。通常は編集する必要はありません。

説明

モデル編集画面におけるこのフローアイテムの名前になります。5.3.15 章の方法でも変更する事が出来ます。

9.1.2 属性設定タブ

属性設定タブでは、図 9.2 のような画面構成になります。

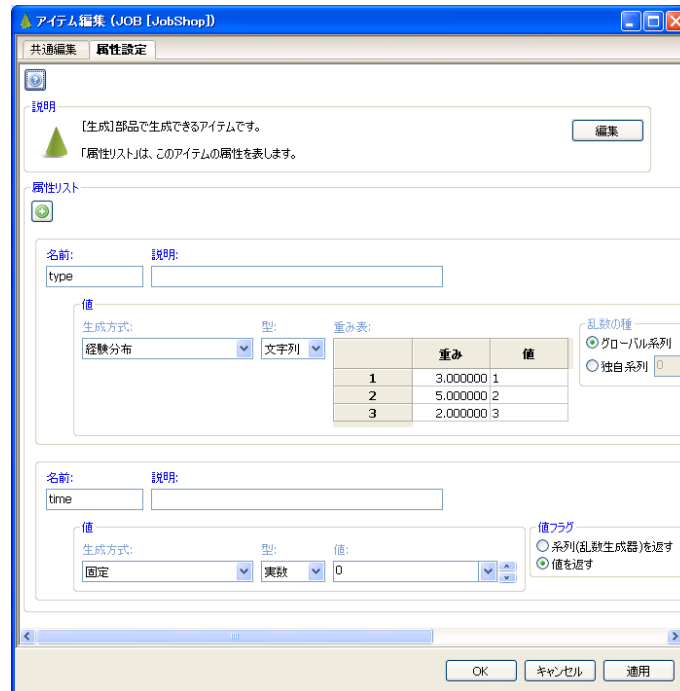


図 9.2: フローアイテム属性設定タブ

上のヘルプボタンを押すと、そのフローアイテムの説明が表示されます。

フローアイテムごとに属性リストとして複数の属性を定義出来ます。

「+」を押すと、属性が追加され末尾に表示されます。

「-」を押すと、その属性が削除されます。

各属性は、以下で構成されます。

名前

フローアイテムの属性の自動生成コード (5.8.1 章) におけるメンバー名になります。

説明

フローアイテムの属性の説明になります。

値

フローアイテムの属性の値を設定します。19 章の任意の乱数を設定する事が出来ます。

値フラグ

「系列を返す」を設定した場合「値」で設定した乱数発生器を返します。「値を返す」を設定した場合、「値」で設定した乱数発生器からひとつの乱数を生成しその結果を返します。

9.2 アイテム (FlowItem)



図 9.3: アイテム

アイテム (図 9.3) は、部品「生成」で生成され、モデル上を流れるアイテムを表します。

例えば、工場内で生産される「製品」、来店する「客」などです。また一定の間隔でアクションを起こすためのトリガーとしても利用できます。

アイテムに関する部品としては、部品「生成」(6.2 章)があります。

属性リスト

アイテムの持つ属性を表します。任意個の異なるデータを属性として設定できます。

第10章 変数

モデル上に配置する変数についてを説明します。

変数とは、連続部品 (6.22 章) による連続シミュレーションによって時間に対し連続的に変化するような変数です。

連続変数と補助変数があります。連続変数は、連続的に変化する変数そのものを示します。補助変数は、連続変数の関数として表せるような変数を示します。

10.1 変数の編集

変数の下のラベルの以外の場所を左ダブルクリック、あるいは、変数の配置されている場所を右クリックして表示されるメニュー (3.5.1 章) から「編集」を選ぶと、変数編集画面が表示されます。

共通設定タブと属性設定タブで構成されます。

10.1.1 共通設定タブ

共通設定タブでは、図 10.1 のような画面構成になります。

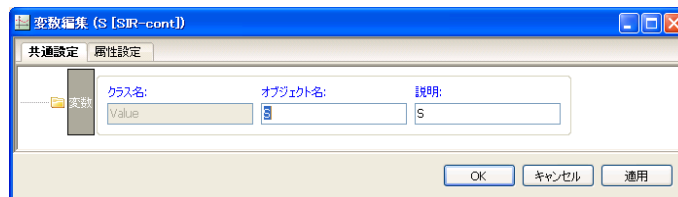


図 10.1: 変数共通設定タブ

オブジェクト名

自動生成コード (5.8.1 章) におけるこの変数のインスタンス名になります。

説明

10.2 連続変数 (Value)

モデル編集画面におけるこの変数の名前になります。5.3.15 章の方法でも変更する事が出来ます。

10.1.2 属性設定タブ

属性設定タブでは、図 10.2 のような画面構成になります。

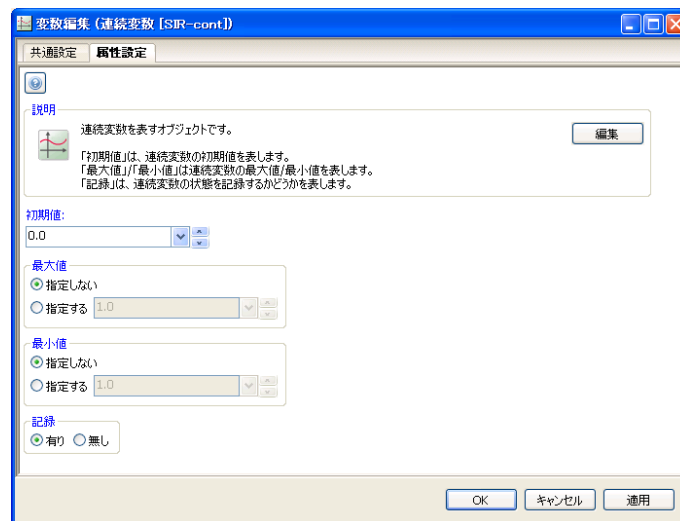


図 10.2: 変数属性設定タブ

連続変数か、補助変数かによって、画面構成は異なりますが、ここで、変数の属性を設定する事ができます。

上のヘルプボタンを押すと、その変数の説明が表示されます。

10.2 連続変数 (Value)



図 10.3: 連続変数

連続変数 (図 10.3) は、連続的に変化する値を表します。

初期値

連続変数の初期値を表します。

最大値

連続変数の最大値を表します。

最小値

連続変数の最小値を表します。

記録

連続変数の状態を記録するかどうかを表します。

10.3 補助変数 (Expression)



図 10.4: 補助変数

補助変数 (図 10.4) は、連続変数の関数です。

参照変数

「式」で使用する連続変数のリストを指定します。

式

「参照変数」に指定した連続変数を使った式を指定します。

記録

補助変数の状態を記録するかどうかを表します。

第11章 エージェント

モデル上に配置するエージェントについて説明します。

エージェントとは、個々の自律的に動作する主体を表現するものです。エージェントには、固定ステップで同期的に動作する同期エージェント、個々のエージェントが非同期に動作する非同期エージェントがあります。エージェントは必ずエージェント集合に属します。

エージェント集合とは、互いに影響を及ぼしあいながら行動する複数のエージェントの集りを表現するものです。エージェント集合は必ず環境に属します。

同期エージェントは、エージェントの一種ですが、特に全てのエージェントの状態が固定タイミングで変化するようなエージェントです。例えば、交通シミュレーションや物理シミュレーションなどで用いられる事が多いです。メリットとしては、エージェント数の増加に対してスケールしやすい事があります。デメリットとしては、より精度の高いシミュレーションにするには、すべてのエージェントのステップ幅を短くする必要があり、(間隔の分散の大きい) 離散イベントの処理は、効率が悪い事などがあります。

非同期エージェントも、エージェントの一種ですが、特に個々のエージェントの状態が独立に離散タイミングで変化するようなエージェントです。例えば、ツイッターの伝播シミュレーション、購買行動シミュレーションなどで用いられる事が多いです。メリットとしては、イベント発生間隔が疎なら、高速な離散イベントシミュレーションができる事があります。デメリットとしては、エージェント数の増加に対して、あまりスケールしない事があります。

11.0.1 属性設定タブ

エージェント編集画面を開くと、最初に属性設定タブが開きます。

属性設定タブでは、図 11.1 のような画面構成になります。

同期エージェントか、非同期エージェントかによって、画面構成は多少異なります。

画面内の編集ボタンを押すと、そのコードを編集するタブが加わり、そのタブが選択状態になります。そのタブ内で、コードを編集する事が出来ます。

上のヘルプボタンを押すと、その部品の説明が表示されます。

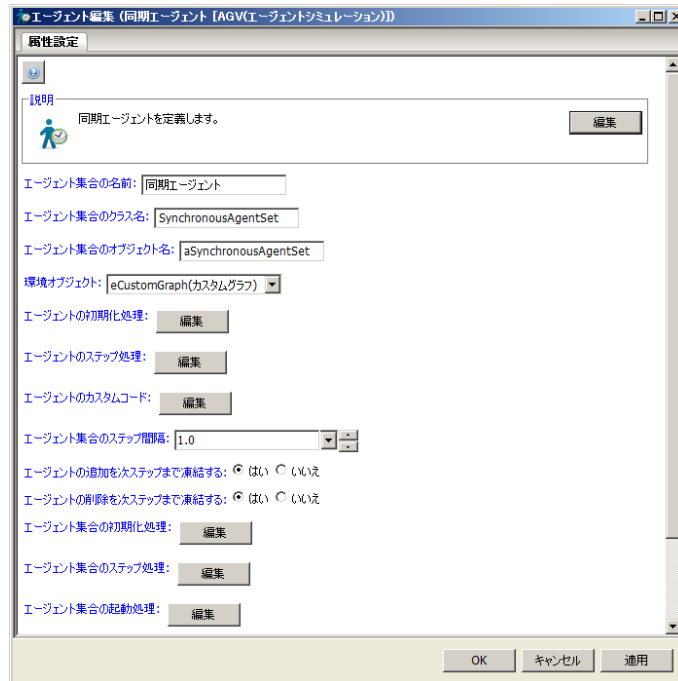


図 11.1: エージェント属性設定タブ

11.1 同期エージェント (SynchronousAgent)



図 11.2: 同期エージェント

「同期エージェント」(図 11.2) は同期エージェントを定義します。
(コード編集についてのより詳細な解説は psim 言語リファレンスマニュアルを参照下さい。)

エージェント集合の名前

エージェント集合の名前を表します。

エージェント集合のクラス名

エージェント集合のクラス名を表します。

エージェント集合のオブジェクト名

エージェント集合のオブジェクト名を表します。

環境オブジェクト

このエージェント集合が属す環境を表します

エージェントの初期化処理

このエージェントの初期化処理を定義します。

エージェントのステップ処理

このエージェントのステップ処理を定義します。次の設定項目を含みます。

強化学習を使用するか

チェックを入れると、強化学習による行動提示をステップ処理中に利用できます。

強化学習モデル

強化学習を利用する際のモデルをあらわします。

エージェント間で学習結果を分けるか

チェックを入れると、強化学習を利用する際に、エージェントごとに学習結果のモデルが分けられます。指定しないとエージェント間で共通の学習モデルが使用されます。

「編集」ボタンを押すと、ステップ処理関数の記述画面が開きます。上記設定項目で、強化学習を使用するようにしていると、関数の最初と最後に強化学習を利用するためのコードが挿入されます。挿入されているコードで定義されている `act` 変数に強化学習により提案された行動値が入っています。 `act` 変数に格納された行動値の各次元の値は `act["次元名"]` という形でアクセスできます。行動値の各次元の値を参考にして、行動に対応した実際の処理を記述して下さい。なお、実際に行った処理に対応する行動が、提案された行動と異なる場合は、対応する行動の値を `act` に設定しなおして下さい。

エージェントのカスタムコード

このエージェントのカスタム処理を定義します。

エージェント集合のステップ間隔

このエージェント集合のステップ間隔を表します。

エージェントの追加を次ステップまで凍結する

指定されていた場合、エージェントの追加処理は次ステップまで凍結されます。

エージェントの削除を次ステップまで凍結する

指定されていた場合、エージェントの削除処理は次ステップまで凍結されます。

エージェント集合の初期化処理

このエージェント集合の初期化処理を定義します。

エージェント集合のステップ処理

このエージェント集合のステップ処理を定義します。

エージェント集合の起動処理

このエージェント集合の起動処理を定義します。

エージェント集合のカスタムコード

このエージェント集合のカスタム処理を定義します。

エージェント集合の可視化

このエージェント集合の可視化処理を定義します。



図 11.3: SFM エージェント

11.2 SFM エージェント (SFMAgent)

「同期エージェント」(図 11.3) は同期エージェントを定義します。
(コード編集についてのより詳細な解説は psim 言語リファレンスマニュアルを参照下さい。)

エージェント集合の名前

エージェント集合の名前を表します。

エージェント集合のクラス名

エージェント集合のクラス名を表します。

エージェント集合のオブジェクト名

エージェント集合のオブジェクト名を表します。

環境オブジェクト

このエージェント集合が属す環境を表します

エージェントの初期化処理

このエージェントの初期化処理を定義します。

エージェントのステップ処理

このエージェントのステップ処理を定義します。次の設定項目を含みます。

強化学習を使用するか

チェックを入れると、強化学習による行動提示をステップ処理中に利用できます。

強化学習モデル

強化学習を利用する際のモデルをあらわします。

エージェント間で学習結果を分けるか

チェックを入れると、強化学習を利用する際に、エージェントごとに学習結果のモデルが分けられます。指定しないとエージェント間で共通の学習モデルが使用されます。

「編集」ボタンを押すと、ステップ処理関数の記述画面が開きます。上記設定項目で、強化学習を使用するようにしていると、関数の最初と最後に強化学習を利用するためのコードが挿入されます。挿入されているコードで定義されている `act` 変数に強化学習により提案された行動値が入っています。 `act` 変数に格納された行動値の各次元の値は `act["次元名"]` という形でアクセスできます。行動値の各次元の値を参考にして、行動に対応した実際の処理を記述して下さい。なお、実際に行った処理に対応する行動が、提案された行動と異なる場合は、対応する行動の値を `act` に設定しなおして下さい。

エージェントの次の経路地点決定処理

このエージェントの次の経路地点を求める処理を定義します。

エージェントのカスタムコード

このエージェントのカスタム処理を定義します。

エージェント集合のステップ間隔

このエージェント集合のステップ間隔を表します。

エージェント集合の初期化処理

このエージェント集合の初期化処理を定義します。

エージェント集合のステップ処理

このエージェント集合のステップ処理を定義します。

エージェント集合の起動処理

このエージェント集合の起動処理を定義します。

エージェント集合のカスタムコード

このエージェント集合のカスタム処理を定義します。

エージェント集合の可視化

このエージェント集合の可視化処理を定義します。

11.3 粒子フィルタエージェント (ParticleFilterAgent)



図 11.4: 粒子フィルタエージェント

「同期エージェント」(図 11.2) は同期エージェントを定義します。
(コード編集についてのより詳細な解説は psim 言語リファレンスマニュアルを参照下さい。)

エージェント集合の名前

エージェント集合の名前を表します。

エージェント集合のクラス名

エージェント集合のクラス名を表します。

エージェント集合のオブジェクト名

エージェント集合のオブジェクト名を表します。

環境オブジェクト

このエージェント集合が属す環境を表します

エージェントの初期化処理

このエージェントの初期化処理を定義します。

エージェントのステップ処理

このエージェントのステップ処理を定義します。次の設定項目を含みます。

強化学習を使用するか

チェックを入れると、強化学習による行動提示をステップ処理中に利用できます。

強化学習モデル

強化学習を利用する際のモデルをあらわします。

エージェント間で学習結果を分けるか

チェックを入れると、強化学習を利用する際に、エージェントごとに学習結果のモデルが分けられます。指定しないとエージェント間で共通の学習モデルが使用されます。

「編集」ボタンを押すと、ステップ処理関数の記述画面が開きます。上記設定項目で、強化学習を使用するようにしていると、関数の最初と最後に強化学習を利用するためのコードが挿入されます。挿入されているコードで定義されている `act` 変数に強化学習により提案された行動値が入っています。`act` 変数に格納された行動値の各次元の値は `act["次元名"]` という形でアクセスできます。行動値の各次元の値を参考にして、行動に対応した実際の処理を記述して下さい。なお、実際に行った処理に対応する行動が、提案された行動と異なる場合は、対応する行動の値を `act` に設定しなおして下さい。

エージェントのカスタムコード

このエージェントのカスタム処理を定義します。

エージェント集合のステップ間隔

このエージェント集合のステップ間隔を表します。

エージェントの追加を次ステップまで凍結する

指定されていた場合、エージェントの追加処理は次ステップまで凍結されます。

エージェントの削除を次ステップまで凍結する

指定されていた場合、エージェントの削除処理は次ステップまで凍結されます。

エージェント集合の初期化処理

このエージェント集合の初期化処理を定義します。

エージェント集合のステップ処理

このエージェント集合のステップ処理を定義します。

エージェント集合の起動処理

このエージェント集合の起動処理を定義します。

エージェント集合のカスタムコード

このエージェント集合のカスタム処理を定義します。

エージェント集合の可視化

このエージェント集合の可視化処理を定義します。

11.4 NW エージェント (NWAgent)



図 11.5: NW エージェント

「同期エージェント」(図 11.3) は同期エージェントを定義します。
(コード編集についてのより詳細な解説は psim 言語リファレンスマニュアルを参照下さい。)

エージェント集合の名前

エージェント集合の名前を表します。

エージェント集合のクラス名

エージェント集合のクラス名を表します。

エージェント集合のオブジェクト名

エージェント集合のオブジェクト名を表します。

環境オブジェクト

このエージェント集合が属す環境を表します

エージェントの初期化処理

このエージェントの初期化処理を定義します。

エージェントのステップ処理

このエージェントのステップ処理を定義します。次の設定項目を含みます。

強化学習を使用するか

チェックを入れると、強化学習による行動提示をステップ処理中に利用できます。

強化学習モデル

強化学習を利用する際のモデルをあらわします。

エージェント間で学習結果を分けるか

チェックを入れると、強化学習を利用する際に、エージェントごとに学習結果のモデルが分けられます。指定しないとエージェント間で共通の学習モデルが使用されます。

「編集」ボタンを押すと、ステップ処理関数の記述画面が開きます。上記設定項目で、強化学習を使用するようにしていると、関数の最初と最後に強化学習を利用するためのコードが挿入されます。挿入されているコードで定義されている `act` 変数に強化学習により提案された行動値が入っています。 `act` 変数に格納された行動値の各次元の値は `act["次元名"]` という形でアクセスできます。行動値の各次元の値を参考にして、行動に対応した実際の処理を記述して下さい。なお、実際に行った処理に対応する行動が、提案された行動と異なる場合は、対応する行動の値を `act` に設定しなおして下さい。

11.5 非同期エージェント (AsynchronousAgent)

エージェントの次の経路地点決定処理

このエージェントの次の経路地点を求める処理を定義します。

エージェントのカスタムコード

このエージェントのカスタム処理を定義します。

エージェント集合のステップ間隔

このエージェント集合のステップ間隔を表します。

エージェント集合の初期化処理

このエージェント集合の初期化処理を定義します。

エージェント集合のステップ処理

このエージェント集合のステップ処理を定義します。

エージェント集合の起動処理

このエージェント集合の起動処理を定義します。

エージェント集合のカスタムコード

このエージェント集合のカスタム処理を定義します。

エージェント集合の可視化

このエージェント集合の可視化処理を定義します。

11.5 非同期エージェント (AsynchronousAgent)



図 11.6: 非同期エージェント

「非同期エージェント」(図 11.6) は非同期エージェントを定義します。
(コード編集についてのより詳細な解説は psim 言語リファレンスマニュアルを参照下さい。)

エージェント集合の名前

エージェント集合の名前を表します。

エージェント集合のクラス名

エージェント集合のクラス名を表します。

エージェント集合のオブジェクト名

エージェント集合のオブジェクト名を表します。

環境オブジェクト

このエージェント集合が属す環境を表します。

エージェントの初期化処理

このエージェントの初期化処理を定義します。

エージェントのプロセス処理

このエージェントのプロセス処理を定義します。

エージェントのカスタムコード

このエージェントのカスタム処理を定義します。

エージェント集合の初期化処理

このエージェント集合の初期化処理を定義します。

エージェント集合の起動処理

このエージェント集合の起動処理を定義します。

エージェント集合のカスタムコード

このエージェント集合のカスタム処理を定義します。

エージェント集合の可視化

このエージェント集合の可視化処理を定義します。

第12章 環境

モデル上に配置する環境について説明します。

環境とは、エージェント行動の基盤となるジオメトリ情報や、全エージェントのテーブルなどを管理するものです。

12.0.1 属性設定タブ

環境編集画面を開くと、最初に属性設定タブが開きます。

属性設定タブでは、図 12.1 のような画面構成になります。

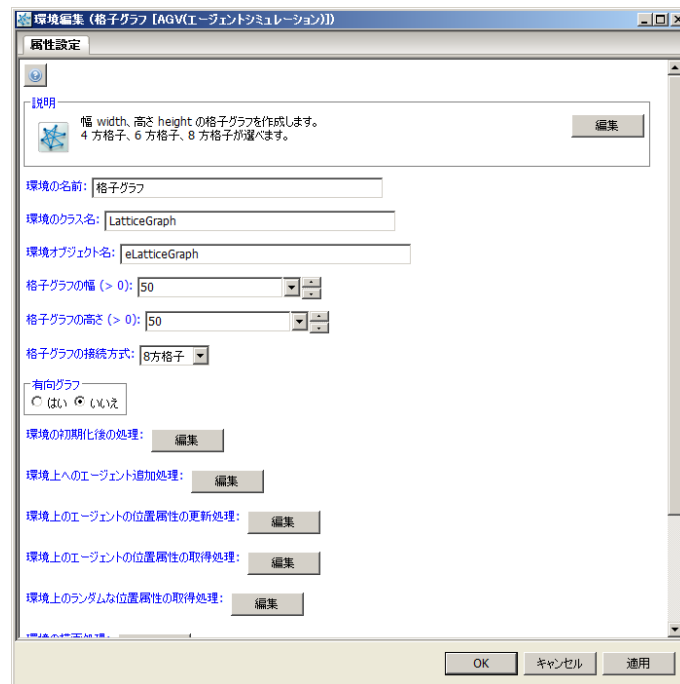


図 12.1: 環境属性設定タブ

環境の種類によって、画面構成は多少異なります。

画面内の編集ボタンを押すと、そのコードを編集するタブが加わり、そのタブが選択状態になります。そのタブ内で、コードを編集する事が出来ます。上のヘルプボタンを押すと、その部品の説明が表示されます。

12.1 SFM 地図 (SFMMMap)



図 12.2: SFM 地図

「SFM 地図」(図 12.2) は、2 次元のユークリッド空間上の、ソーシャルフォースモデル環境を作成します。SFM 地図エディタ上で、地図を編集 (13 章) 出来ます。

(コード編集についてのより詳細な解説は psim 言語リファレンスマニュアルを参照下さい。)

環境の名前

環境の名前を表します。

環境のクラス名

環境のクラス名を表します。

環境オブジェクト名

環境のオブジェクト名を表します。

地図

「地図編集」を選択すると、SFM 地図エディタが起動します。
SFM 地図エディタ上で、地図を編集 (13 章) 出来ます。

相互作用の強さ (N)、相互作用の範囲 (m)、弾性係数 (kg/s²)、散逸係数 (kg/ms)、最大影響半径 (m)、経路再探索間隔 (s)、最適速度 (m/s)、最高速度 (m/s)、歩行者の半径 (m)、加速時間 (s)、体重 (kg)、外力の変動係数

ソーシャルフォースモデルのパラメータを表します。

歩行者半径表示、速度ベクトル表示、歩行者間外力表示、障害物外力表示、経路グラフ表示、Corridor Map、背景画像、ユーザー定義領域

エージェントフレーム上の表示方式を指定します。「環境上のエージェントの描画処理」を修正する事で、細かな調整も可能です。

環境の初期化後の処理

この環境の初期化処理を定義します。

環境の描画処理

この環境の描画処理を定義します。

環境のエージェントの描画処理

この環境上のエージェントの描画処理を定義します。

環境のカスタムコード

この環境のカスタムコードを定義します。

12.2 SFM 環境 (SFMEnvironment)



図 12.3: SFM 環境

「SFM 環境」(図 12.3) は、2次元のユークリッド空間上の、ソーシャルフォースモデル環境を作成します。x 座標は x_0 から x_1 までをとり、y 座標は y_0 から y_1 までをとります。

(コード編集についてのより詳細な解説は psim 言語リファレンスマニュアルを参照下さい。)

環境の名前

環境の名前を表します。

環境のクラス名

環境のクラス名を表します。

環境オブジェクト名

環境のオブジェクト名を表します。

x_0 , x_1

x 座標の範囲を表します。

y0, y1

y 座標の範囲を表します。

相互作用の強さ $A(N)$ 、相互作用の範囲 $B(m)$ 、弾性係数 $k(kg/s^2)$ 、散逸係数 $\kappa(kg/ms)$ 、最大影響半径 (m) 、経路再探索間隔 (s) 、最適速度 (m/s) 、最高速度 (m/s) 、歩行者の半径 (m) 、加速時間 (s) 、体重 (kg) 、外力の変動係数

ソーシャルフォースモデルのパラメータを表します。

歩行者半径表示、速度ベクトル表示、歩行者間外力表示、障害物外力表示、経路グラフ表示

エージェントフレーム上の表示方式を指定します。「環境上のエージェントの描画処理」を修正する事で、細かな調整も可能です。

環境の初期化後の処理

この環境の初期化処理を定義します。

環境の描画処理

この環境の描画処理を定義します。

環境のエージェントの描画処理

この環境上のエージェントの描画処理を定義します。

環境のカスタムコード

この環境のカスタムコードを定義します。



図 12.4: PowerlawCluster グラフ

12.3 PowerlawCluster グラフ (PowerlawClusterRandomGraph)

「PowerlawCluster グラフ」(図 12.4) は、スケールフリー性 (次数分布のべき乗則) を持ったネットワークを作成します。m 個のノードから空のグラフから開始し、ノード数が n になるまで成長させるのは Barabasi-Albert モデルと同様ですが、ノードの追加時に、確率 p で、triangle を生成します。グラフは無向グラフになります。

(コード編集についてのより詳細な解説は psim 言語リファレンスマニュアルを参照下さい。)

環境の名前

環境の名前を表します。

環境のクラス名

環境のクラス名を表します。

環境オブジェクト名

環境のオブジェクト名を表します。

ノード数

n を表します。

各ステップで追加するエッジ数

m を表します。

triangle を生成する確率

p を表します。

乱数の種

乱数の種を表します。グローバル系列の場合はグローバル系列から乱数の種を初期化します。

環境の初期化後の処理

この環境の初期化処理を定義します。

環境上へのエージェント追加処理

この環境へエージェントを追加した時の処理を定義します。

環境上のエージェント位置属性の更新処理

この環境のエージェントの位置属性を更新した時の処理を定義します。

環境上のエージェント位置属性の取得処理

この環境のエージェントの位置属性を取得する処理を定義します。

環境上のランダムな位置属性の取得処理

この環境のエージェントのランダムな位置属性を取得する処理を定義します。

環境の描画処理

この環境の描画処理を定義します。

環境のエージェントの描画処理

この環境上のエージェントの描画処理を定義します。

環境のカスタムコード

この環境のカスタムコードを定義します。



図 12.5: NW 地図

12.4 NW 地図 (NWMap)

「SFM 地図」(図 12.2) は、2次元のユークリッド空間上の、ソーシャルフォースモデル環境を作成します。SFM 地図エディタ上で、地図を編集 (13章) 出来ます。

(コード編集についてのより詳細な解説は psim 言語リファレンスマニュアルを参照下さい。)

環境の名前

環境の名前を表します。

環境のクラス名

環境のクラス名を表します。

環境オブジェクト名

環境のオブジェクト名を表します。

地図

「地図編集」を選択すると、SFM 地図エディタが起動します。
SFM 地図エディタ上で、地図を編集 (13章) 出来ます。

相互作用の強さ (N)、相互作用の範囲 (m)、弾性係数 (kg/s²)、散逸係数 (kg/ms)、最大影響半径 (m)、経路再探索間隔 (s)、最適速度 (m/s)、最高速度 (m/s)、歩行者の半径 (m)、加速時間 (s)、体重 (kg)、外力の変動係数

ソーシャルフォースモデルのパラメータを表します。

歩行者半径表示、速度ベクトル表示、歩行者間外力表示、障害物外力表示、経路グラフ表示、Corridor Map、背景画像、ユーザー定義領域

エージェントフレーム上の表示方式を指定します。「環境上のエージェントの描画処理」を修正する事で、細かな調整も可能です。

環境の初期化後の処理

この環境の初期化処理を定義します。

環境の描画処理

この環境の描画処理を定義します。

環境のエージェントの描画処理

この環境上のエージェントの描画処理を定義します。

環境のカスタムコード

この環境のカスタムコードを定義します。

12.5 格子グラフ (LatticeGraph)



図 12.6: 格子グラフ

「格子グラフ」(図 12.6) は、幅 width、高さ height の格子グラフを作成します。4 方格子、6 方格子、8 方格子が選べます。

(コード編集についてのより詳細な解説は psim 言語リファレンスマニュアルを参照下さい。)

環境の名前

環境の名前を表します。

環境のクラス名

環境のクラス名を表します。

環境オブジェクト名

環境のオブジェクト名を表します。

格子グラフの幅

width を表します。

格子グラフの高さ

height を表します。

格子グラフの接続方式

4 方格子、6 方格子、8 方格子を表します。

有向グラフ

有向グラフか否か指定します。

環境の初期化後の処理

この環境の初期化処理を定義します。

環境上へのエージェント追加処理

この環境へエージェントを追加した時の処理を定義します。

環境上のエージェント位置属性の更新処理

この環境のエージェントの位置属性を更新した時の処理を定義します。

環境上のエージェント位置属性の取得処理

この環境のエージェントの位置属性を取得する処理を定義します。

環境上のランダムな位置属性の取得処理

この環境のエージェントのランダムな位置属性を取得する処理を定義します。

環境の描画処理

この環境の描画処理を定義します。

環境のエージェントの描画処理

この環境上のエージェントの描画処理を定義します。

環境のカスタムコード

この環境のカスタムコードを定義します。

12.6 GraphML フォーマットグラフ (GraphML-FormatGraph)



図 12.7: GraphML フォーマットグラフ

「GraphML フォーマットグラフ」(図 12.7) は、GraphML フォーマットのグラフをファイルからグラフを作成します。

GraphML フォーマットについては、22.1.2 章を参照して下さい。

(コード編集についてのより詳細な解説は psim 言語リファレンスマニュアルを参照下さい。)

環境の名前

環境の名前を表します。

環境のクラス名

環境のクラス名を表します。

環境オブジェクト名

環境のオブジェクト名を表します。

グラフファイル (GraphML 形式)

「グラフファイルの取り込み」を押すと、GraphML ファイルのインポートダイアログが開くので、ここで取り込みたい GraphML ファイルを選択します。

環境の初期化後の処理

この環境の初期化処理を定義します。

環境上へのエージェント追加処理

この環境へエージェントを追加した時の処理を定義します。

環境上のエージェント位置属性の更新処理

この環境のエージェントの位置属性を更新した時の処理を定義します。

環境上のエージェント位置属性の取得処理

この環境のエージェントの位置属性を取得する処理を定義します。

環境上のランダムな位置属性の取得処理

この環境のエージェントのランダムな位置属性を取得する処理を定義します。

環境の描画処理

この環境の描画処理を定義します。

環境のエージェントの描画処理

この環境上のエージェントの描画処理を定義します。

環境のカスタムコード

この環境のカスタムコードを定義します。

12.7 GNP グラフ (GNPRandomGraph)



図 12.8: GNP グラフ

「GNP グラフ」(図 12.8) は、ノード数 n のランダムグラフを作成します。おのおののエッジは確率 p で生成されます。

(コード編集についてのより詳細な解説は psim 言語リファレンスマニュアルを参照下さい。)

環境の名前

環境の名前を表します。

環境のクラス名

環境のクラス名を表します。

環境オブジェクト名

環境のオブジェクト名を表します。

ノード数

n を表します。

エッジ生成確率

p を表します。

有向グラフ

有向グラフか否か指定します。

乱数の種

乱数の種を表します。グローバル系列の場合はグローバル系列から乱数の種を初期化します。

環境の初期化後の処理

この環境の初期化処理を定義します。

環境上へのエージェント追加処理

この環境へエージェントを追加した時の処理を定義します。

環境上のエージェント位置属性の更新処理

この環境のエージェントの位置属性を更新した時の処理を定義します。

環境上のエージェント位置属性の取得処理

この環境のエージェントの位置属性を取得する処理を定義します。

環境上のランダムな位置属性の取得処理

この環境のエージェントのランダムな位置属性を取得する処理を定義します。

環境の描画処理

この環境の描画処理を定義します。

環境のエージェントの描画処理

この環境上のエージェントの描画処理を定義します。

環境のカスタムコード

この環境のカスタムコードを定義します。

12.8 GNM グラフ (GNMRandomGraph)



図 12.9: GNM グラフ

「GNM グラフ」(図 12.9) は、ノード数 n , エッジ数 m のランダムグラフを作成します。ただし、 m が完全グラフのエッジ数を越えた場合は、完全グラフを返します。

(コード編集についてのより詳細な解説は psim 言語リファレンスマニュアルを参照下さい。)

環境の名前

環境の名前を表します。

環境のクラス名

環境のクラス名を表します。

環境オブジェクト名

環境のオブジェクト名を表します。

ノード数

n を表します。

エッジ数

m を表します。

有向グラフ

有向グラフか否か指定します。

乱数の種

乱数の種を表します。グローバル系列の場合はグローバル系列から乱数の種を初期化します。

環境の初期化後の処理

この環境の初期化処理を定義します。

環境上へのエージェント追加処理

この環境へエージェントを追加した時の処理を定義します。

環境上のエージェント位置属性の更新処理

この環境のエージェントの位置属性を更新した時の処理を定義します。

環境上のエージェント位置属性の取得処理

この環境のエージェントの位置属性を取得する処理を定義します。

環境上のランダムな位置属性の取得処理

この環境のエージェントのランダムな位置属性を取得する処理を定義します。

環境の描画処理

この環境の描画処理を定義します。

環境のエージェントの描画処理

この環境上のエージェントの描画処理を定義します。

環境のカスタムコード

この環境のカスタムコードを定義します。

12.9 GEXF フォーマットグラフ (GEXFFormatGraph)



図 12.10: GEXF フォーマットグラフ

「GEXF フォーマットグラフ」(図 12.10) は、GEXF フォーマットのグラフをファイルからグラフを作成します。

GEXF フォーマットについては、[22.1.1](#) 章を参照して下さい。

(コード編集についてのより詳細な解説は `psim` 言語リファレンスマニュアルを参照下さい。)

環境の名前

環境の名前を表します。

環境のクラス名

環境のクラス名を表します。

環境オブジェクト名

環境のオブジェクト名を表します。

グラフファイル (GEXF 形式)

「グラフファイルの取り込み」を押すと、GEXF ファイルのインポートダイアログが開くので、ここで取り込みたい GEXF ファイルを選択します。

環境の初期化後の処理

この環境の初期化処理を定義します。

環境上へのエージェント追加処理

この環境へエージェントを追加した時の処理を定義します。

環境上のエージェント位置属性の更新処理

この環境のエージェントの位置属性を更新した時の処理を定義します。

環境上のエージェント位置属性の取得処理

この環境のエージェントの位置属性を取得する処理を定義します。

環境上のランダムな位置属性の取得処理

この環境のエージェントのランダムな位置属性を取得する処理を定義します。

環境の描画処理

この環境の描画処理を定義します。

環境のエージェントの描画処理

この環境上のエージェントの描画処理を定義します。

環境のカスタムコード

この環境のカスタムコードを定義します。



図 12.11: ユークリッド 2D

12.10 ユークリッド 2D(Euclid2D)

「ユークリッド 2D」(図 12.11) は、2次元のユークリッド空間を作成します。x 座標は x_0 から x_1 までをとり、y 座標は y_0 から y_1 までをとります。(コード編集についてのより詳細な解説は psim 言語リファレンスマニュアルを参照下さい。)

環境の名前

環境の名前を表します。

環境のクラス名

環境のクラス名を表します。

環境オブジェクト名

環境のオブジェクト名を表します。

x_0, x_1

x 座標の範囲を表します。

y_0, y_1

y 座標の範囲を表します。

環境の初期化後の処理

この環境の初期化処理を定義します。

環境上へのエージェント追加処理

この環境へエージェントを追加した時の処理を定義します。

環境上のエージェント位置属性の更新処理

この環境のエージェントの位置属性を更新した時の処理を定義します。

環境上のエージェント位置属性の取得処理

この環境のエージェントの位置属性を取得する処理を定義します。

環境上のランダムな位置属性の取得処理

この環境のエージェントのランダムな位置属性を取得する処理を定義します。

環境の描画処理

この環境の描画処理を定義します。

環境のエージェントの描画処理

この環境上のエージェントの描画処理を定義します。

環境のカスタムコード

この環境のカスタムコードを定義します。

12.11 環境 (Environment)



図 12.12: 環境

「環境」(図 12.12) は、空の環境を作成します。

(コード編集についてのより詳細な解説は psim 言語リファレンスマニュアルを参照下さい。)

環境の名前

環境の名前を表します。

環境のクラス名

環境のクラス名を表します。

環境オブジェクト名

環境のオブジェクト名を表します。

環境の初期化後の処理

この環境の初期化処理を定義します。

環境上へのエージェント追加処理

この環境へエージェントを追加した時の処理を定義します。

環境上のエージェント位置属性の更新処理

この環境のエージェントの位置属性を更新した時の処理を定義します。

環境上のエージェント位置属性の取得処理

この環境のエージェントの位置属性を取得する処理を定義します。

環境上のランダムな位置属性の取得処理

この環境のエージェントのランダムな位置属性を取得する処理を定義します。

環境の描画処理

この環境の描画処理を定義します。

環境のエージェントの描画処理

この環境上のエージェントの描画処理を定義します。

環境のカスタムコード

この環境のカスタムコードを定義します。



図 12.13: カスタムグラフ

12.12 カスタムグラフ (CustomGraph)

「カスタムグラフ」(図 12.13) は、空のグラフを作成します。
(コード編集についてのより詳細な解説は psim 言語リファレンスマニュアルを参照下さい。)

環境の名前

環境の名前を表します。

環境のクラス名

環境のクラス名を表します。

環境オブジェクト名

環境のオブジェクト名を表します。

有向グラフ

有向グラフか否か指定します。

環境の初期化後の処理

この環境の初期化処理を定義します。

環境上へのエージェント追加処理

この環境へエージェントを追加した時の処理を定義します。

環境上のエージェント位置属性の更新処理

この環境のエージェントの位置属性を更新した時の処理を定義します。

環境上のエージェント位置属性の取得処理

この環境のエージェントの位置属性を取得する処理を定義します。

環境上のランダムな位置属性の取得処理

この環境のエージェントのランダムな位置属性を取得する処理を定義します。

環境の描画処理

この環境の描画処理を定義します。

環境のエージェントの描画処理

この環境上のエージェントの描画処理を定義します。

環境のカスタムコード

この環境のカスタムコードを定義します。

12.13 完全グラフ (CompleteGraph)



図 12.14: 完全グラフ

「完全グラフ」(図 12.14) は、ノード数 n の完全グラフを作成します。
(コード編集についてのより詳細な解説は psim 言語リファレンスマニュアルを参照下さい。)

環境の名前

環境の名前を表します。

環境のクラス名

環境のクラス名を表します。

環境オブジェクト名

環境のオブジェクト名を表します。

ノード数

n を表します。

有向グラフ

有向グラフか否か指定します。

環境の初期化後の処理

この環境の初期化処理を定義します。

環境上へのエージェント追加処理

この環境へエージェントを追加した時の処理を定義します。

環境上のエージェント位置属性の更新処理

この環境のエージェントの位置属性を更新した時の処理を定義します。

環境上のエージェント位置属性の取得処理

この環境のエージェントの位置属性を取得する処理を定義します。

環境上のランダムな位置属性の取得処理

この環境のエージェントのランダムな位置属性を取得する処理を定義します。

環境の描画処理

この環境の描画処理を定義します。

環境のエージェントの描画処理

この環境上のエージェントの描画処理を定義します。

環境のカスタムコード

この環境のカスタムコードを定義します。



図 12.15: BarabasiAlbert グラフ

12.14 BarabasiAlbert グラフ (BarabasiAlbertRandomGraph)

「BarabasiAlbert グラフ」(図 12.15) は、Barabasi-Albert モデルに従ったスケールフリーネットワークを作成します。m 個のノードから空のグラフから開始し、ノード数が n になるまで成長させます。ノードを追加する時、既に存在する m 個のノードにエッジを張りますが、この時、エッジが張られる確率は、それぞれのノードのその時点での度数に比例します。グラフは無向グラフになります。

(コード編集についてのより詳細な解説は psim 言語リファレンスマニュアルを参照下さい。)

環境の名前

環境の名前を表します。

環境のクラス名

環境のクラス名を表します。

環境オブジェクト名

環境のオブジェクト名を表します。

ノード数

n を表します。

各ステップで追加するエッジ数

m を表します。

乱数の種

乱数の種を表します。グローバル系列の場合はグローバル系列から乱数の種を初期化します。

環境の初期化後の処理

この環境の初期化処理を定義します。

環境上へのエージェント追加処理

この環境へエージェントを追加した時の処理を定義します。

環境上のエージェント位置属性の更新処理

この環境のエージェントの位置属性を更新した時の処理を定義します。

環境上のエージェント位置属性の取得処理

この環境のエージェントの位置属性を取得する処理を定義します。

環境上のランダムな位置属性の取得処理

この環境のエージェントのランダムな位置属性を取得する処理を定義します。

環境の描画処理

この環境の描画処理を定義します。

環境のエージェントの描画処理

この環境上のエージェントの描画処理を定義します。

環境のカスタムコード

この環境のカスタムコードを定義します。

第13章 SFM地図エディタ

SFM 地図エディタについて説明します。

SFM 地図エディタは、SFM(ソーシャルフォースモデル)の地図環境を編集するエディタです。

SFM 地図エディタは、SFM 地図環境内から起動する事が出来ます。

13.1 SFM 地図エディタの概要

起動画面は、図 13.1 のような画面構成になっています。

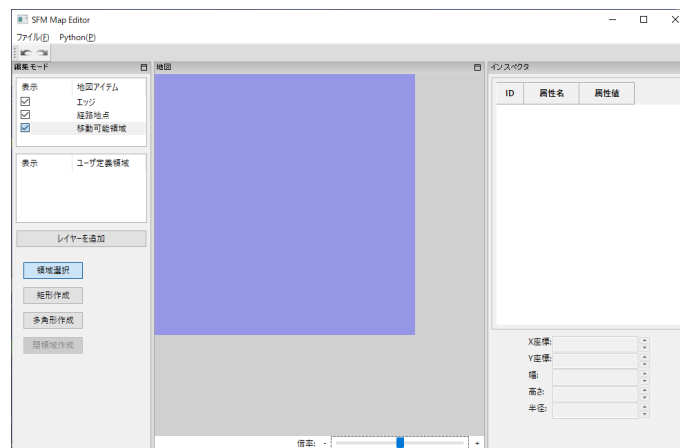


図 13.1: SFM 地図エディタ

左側に編集モードタブがあり、中央に地図エディタタブがあり、右側にインスペクタタブがあります。

編集モードには、以下のモードがあります。

- 移動可能領域操作モード
- 経路地点操作モード
- エッジ操作モード
- 画像操作モード

- 文字操作モード
- ユーザー定義領域操作モード

適切なモードを選択して、地図エディタにて、地図を編集します。

大きくわけて、SFM 地図環境は 4 層構造になっています。

第一層は、環境の移動可能な 2 次元空間を多角形の集合として表わします (移動可能領域)。

第二層には、移動可能な空間上に、SFM の経路地点とエッジが乗っています (経路地点とエッジ)。

第三層には、シミュレーション時に参照できる独自の領域があります (ユーザー定義領域)。

第四層には、シミュレーション時に表示するための要素があります (画像と文字)。

13.2 移動可能領域操作モード

移動可能領域操作モードは、移動可能な 2 次元空間を編集するモードです。

領域の基本単位は、多角形であり、多角形を組み合わせる事で、様々な形状を表現します。

白い領域は移動可能空間、青い領域は移動不可空間 (壁) を示しています。

領域を指定すると、例えば、図 13.2 のようになります。

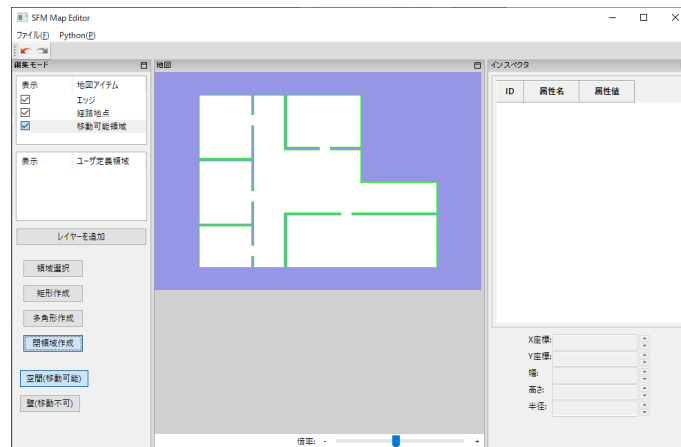


図 13.2: 領域設定後

領域操作モードには、以下のツールがあります。

- 領域選択
- 矩形作成

- 多角形作成
- 閉領域作成

13.2.1 領域選択ツール

領域選択ツールは、領域の基本単位 (多角形) を操作するツールです。

領域上で、左クリックすると、その領域が選択されます。

Shift をクリックしながら、領域をクリックすると、その領域も選択状態に加わります。

領域上で、左クリックを押しながらドラッグすると、その領域が移動します。

選択やドラッグといった操作を行うと、右側のインスペクタに情報が表示されます。

領域上で、右クリックすると、コンテキストメニューが表示されます。

削除

選択されている領域が削除されます。

回転

選択されている領域が回転します。

拡大縮小

選択されている領域が拡大縮小します。

グループ化

選択されている領域がグループ化します。

グループ解除

選択されている領域をグループ解除します。

背面に移動

選択されている領域を背面に移動します。

最背面に移動

選択されている領域を最背面に移動します。

前面に移動

選択されている領域を前面に移動します。

最前面に移動

選択されている領域を最前面に移動します。

13.2.2 矩形作成ツール

矩形作成ツールは、矩形の領域を作成するツールです。

このツールを選択すると、編集モードタブの下部に、「空間 (移動可能)」と「壁 (移動不可)」が表示されます。

空間 (移動可能) が選択されている時は、空間を作成し、壁 (移動不可) が選択されている時は、壁を作成します。

地図エディタタブにてドラッグする事で、選択している種類の矩形領域が作成されます。

13.2.3 多角形作成ツール

多角形作成ツールは、多角形の領域を作成するツールです。

編集モードタブの下部に、「空間 (移動可能)」と「壁 (移動不可)」があります。

空間 (移動可能) が選択されている時は、空間を作成し、壁 (移動不可) が選択されている時は、壁を作成します。

地図エディタタブにてクリックすると、最初の多角形の頂点が作成されます。続いてクリックする度に、多角形の頂点が追加されていきます。辺が交差するような多角形は作成できません。右クリックすると最後の点が削除されます。3 点以上頂点が作成されている状態で最初の頂点の丸にマウスを移動すると、領域が赤く表示され、そこでクリックすると、多角形が確定されます。

注意: 多角形選択中はメニュー操作等ができなくなります。多角形作成をキャンセルしたい場合、右クリックを連続して押して下さい。

13.2.4 閉領域作成ツール

閉領域作成ツールは、背景の DXF 情報に基いて閉領域を作成するツールです。

13.3 経路地点操作モード

このツールは、背景に DXF が設定されている時のみに選択可能です。背景に DXF が設定されていない時は選択出来ません。

背景に DXF が設定されている場合、背景に DXF のエッジが表示されています。その DXF 上で任意の点をクリックすると、その領域が含まれる閉領域が多角形領域として認識されます。

領域の認識は DXF のバージョンによっては誤動作する場合もあります。認識可能な DXF エンティティは、LINE, LWPOLYLINE, CIRCLE, ARC のみです。CIRCLE, ARC に関しては、読み込み時に LINE に分割されます。認識方法としては、クリック点から下方に垂線を降ろし、その点から反時計回りに、一周した領域を認識します。領域内部に領域があった場合でも、それらも含めて、領域として認識します。微小な間隔のノード群は同一点と判定します。

オリジナルの DXF ファイルに領域認識に必要な情報がある場合でも、それらの線分も境界と判断してしまいます。その場合、DXF を編集する外部ツール等を利用して、必要のないレイヤーを削除などの前処理を行う事を推奨します。あるいは、背景情報はあくまでも参考として、その上に矩形作成ツール、多角形作成ツールを利用して、領域を作成していく方法もあります。

13.3 経路地点操作モード

経路点操作モードは、移動可能領域上の経路地点を編集するモードです。経路地点を追加すると、例えば、図 13.3 のようになります。

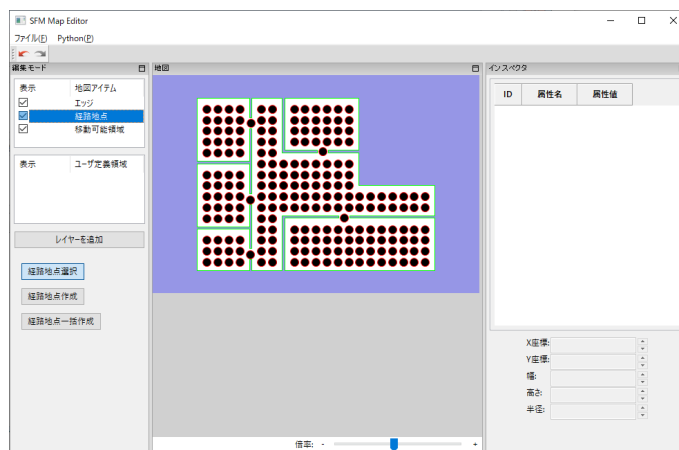


図 13.3: 経路地点追加後

経路点操作モードには、以下のツールがあります。

- 経路地点選択

- 経路地点作成
- 経路地点一括作成

13.3.1 経路地点選択ツール

経路地点選択ツールは、経路地点を操作するツールです。

経路地点上で、左クリックすると、その経路地点が選択されます。

Shift をクリックしながら、経路地点をクリックすると、その経路地点も選択状態に加わります。

経路地点上で、左クリックを押しながらドラッグすると、その経路地点が移動します。

選択やドラッグといった操作を行うと、右側のインスペクタに情報が表示されます。

経路地点上で、右クリックすると、コンテキストメニューが表示されます。

削除

選択されている経路地点が削除されます。

全経路地点を選択

全経路地点が選択されます。

エラー経路地点を選択

エラー経路地点が選択されます。

13.3.2 経路地点作成ツール

経路地点作成ツールは、経路地点を作成するツールです。

このツールを選択すると、編集モードタブの下部に、「半径 (m)」を入力するテキストボックスが表示されます。

地図エディタタブにてクリックすると、その場所に、指定された半径の経路地点が作成されます。

13.3.3 経路地点一括作成ツール

経路地点作成ツールは、経路地点を一括作成するツールです。

このツールを選択すると、編集モードタブの下部に、「半径 (m)」と「間隔 (m)」を入力するテキストボックスが表示されます。

地図エディタタブにてドラッグして矩形領域を指定すると、その領域内に、指定された間隔で指定された半径の経路地点が一括作成されます。

13.3.4 属性管理テーブル

経路地点は属性を持ちます。属性は「属性名」と「属性値」のペアの集合として表現されます。

編集モードの「経路地点」で右クリックすると、コンテキストメニューが表示され、属性管理テーブルを表示できます。

属性管理テーブルでは、属性を追加したり、属性を右クリックしてコンテキストメニューから属性を操作できます。

属性はインスペクタに表示されます。

13.4 エッジ操作モード

エッジ操作モードは、移動可能領域上の経路地点間を結ぶエッジを編集するモードです。

エッジを追加すると、例えば、図 13.4 のようになります。

エッジ操作モードには、以下のツールがあります。

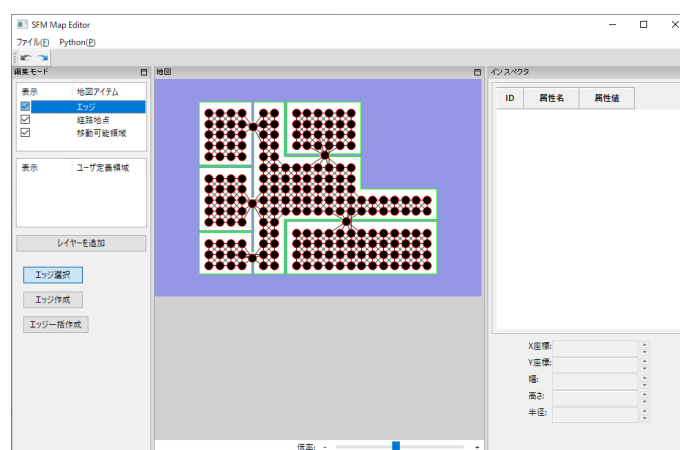


図 13.4: 経路地点追加後

- エッジ選択
- エッジ作成
- エッジ一括作成

13.4.1 エッジ選択ツール

エッジ選択ツールは、エッジを操作するツールです。

エッジ上で、左クリックすると、そのエッジが選択されます。

Shift をクリックしながら、エッジをクリックすると、そのエッジも選択状態に加わります。

選択を行うと、右側のインスペクタに情報が表示されます。

エッジ上で、右クリックすると、コンテキストメニューが表示されます。

削除

選択されているエッジが削除されます。

全エッジを選択

全エッジが選択されます。

エラーエッジを選択

エラーエッジが選択されます。

13.4.2 エッジ作成ツール

経路地点作成ツールは、エッジを作成するツールです。

地図エディタタブにて始点の経路地点からドラッグし、終点の経路地点でボタンを離すとその経路地点間に、エッジが作成されます。

13.4.3 エッジ一括作成ツール

エッジ作成ツールは、エッジを一括作成するツールです。

このツールを選択すると、編集モードタブの下部に、「最大距離 (m)」を入力するテキストボックスが表示されます。

地図エディタタブにてドラッグして矩形領域を指定すると、その領域内の、指定された距離内の全ての経路地点間のエッジが一括作成されます。

13.4.4 属性管理テーブル

エッジは属性を持ちます。属性は「属性名」と「属性値」のペアの集合として表現されます。

編集モードの「エッジ」で右クリックすると、コンテキストメニューが表示され、属性管理テーブルを表示できます。

属性管理テーブルでは、属性を追加したり、属性を右クリックしてコンテキストメニューから属性を操作できます。

属性はインスペクタに表示されます。

13.5 画像操作モード

画像操作モードは、画像を配置するモードです。
配置した画像は S4 シミュレーションでも表示されます。
画像操作モードには、以下のツールがあります。

- 画像選択
- 画像作成

13.5.1 画像選択ツール

画像選択ツールは、画像を操作するツールです。
画像上で、左クリックすると、その画像が選択されます。
Shift をクリックしながら、画像をクリックすると、その画像も選択状態に加わります。

画像上で、左クリックを押しながらドラッグすると、その画像が移動します。
選択やドラッグといった操作を行うと、右側のインスペクタに情報が表示されます。

領域上で、右クリックすると、コンテキストメニューが表示されます。

削除

選択されている画像が削除されます。

回転

選択されている画像が回転します。

拡大縮小

選択されている画像が拡大縮小します。

背面に移動

選択されている画像を背面に移動します。

最背面に移動

選択されている画像を最背面に移動します。

前面に移動

選択されている画像を前面に移動します。

最前面に移動

選択されている画像を最前面に移動します。

13.5.2 画像作成ツール

画像作成ツールは、画像を作成するツールです。

このツールを選択すると、編集モードタブの下部に、「画像ファイルを選択...」と「倍率」が表示されます。

「画像ファイルを選択...」にて画像ファイルを選択の上、マウスを地図上に置くと、配置画像が追隨して表示されます。クリックで実際に配置できます。「倍率」は配置時の画像の大きさの係数を表します。

13.6 文字操作モード

文字操作モードは、文字を配置するモードです。

配置した文字は S4 シミュレーションでも表示されます。

文字操作モードには、以下のツールがあります。

- 文字選択
- 文字作成

13.6.1 文字選択ツール

文字選択ツールは、文字を操作するツールです。

文字上で、左クリックすると、その文字が選択されます。

Shift をクリックしながら、文字をクリックすると、その文字も選択状態に加わります。

文字上で、左クリックを押しながらドラッグすると、その文字が移動します。

選択やドラッグといった操作を行うと、右側のインスペクタに情報が表示されます。

領域上で、右クリックすると、コンテキストメニューが表示されます。

削除

選択されている文字が削除されます。

色変更

選択されている文字の色を変更します。

回転

選択されている文字が回転します。

背面に移動

選択されている文字を背面に移動します。

最背面に移動

選択されている文字を最背面に移動します。

前面に移動

選択されている文字を前面に移動します。

最前面に移動

選択されている文字を最前面に移動します。

13.6.2 文字作成ツール

文字作成ツールは、文字を作成するツールです。

このツールを選択し、地図上で左クリックすると、その位置にテキストボックスが現れます（Enter で複数行入力できます）。テキストボックス外でクリックすると入力が終わり、文字が生成されます。

13.7 レイヤー

地図エディタは0個以上のレイヤーを持ちます。レイヤーは後述のユーザー定義領域を保持するための概念です。すべてのユーザー定義領域はいずれかのレイヤーに所属します。

経路地点やエッジはそれぞれ1つの属性管理テーブルが紐ついていましたが、ユーザー定義領域はレイヤーごとに異なる属性管理テーブルが紐つきます。

レイヤーは「レイヤーを追加」ボタンで追加できます。

レイヤーの上で右クリックすると、コンテキストメニューが表示されます。

属性管理テーブルを表示

選択されているレイヤーの属性管理テーブルを表示します。

レイヤー名を変更

選択されているレイヤーの名前を変更します。

削除

選択されているレイヤーを削除します。

背面に移動

選択されているレイヤーを背面に移動します。

前面に移動

選択されているレイヤーを前面に移動します。

13.8 ユーザー定義領域操作モード

ユーザー定義領域操作モードは、シミュレーション時に参照できる独自の領域を編集するモードです。

領域の基本単位は、多角形であり、多角形を組み合わせる事で、様々な形状を表現します。

内部が半透明の色で塗られている領域は正の領域、内部が透明 (枠のみ) の領域は負の領域と呼びます。

同一のレイヤーに所属するユーザー定義領域は同じ色を持ちます。色は自動的に決定されます。

領域を指定すると、例えば、図 13.5 のようになります。

領域操作モードには、以下のツールがあります。

- 領域選択
- 矩形作成
- 多角形作成
- 閉領域作成

13.8 ユーザー定義領域操作モード

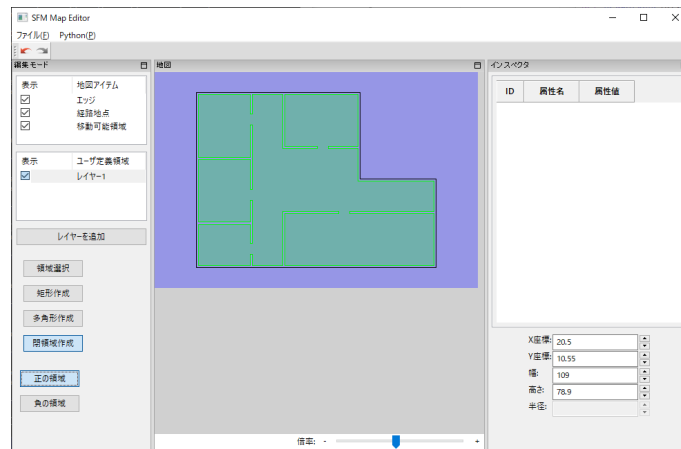


図 13.5: 領域設定後

13.8.1 領域選択ツール

領域選択ツールは、領域の基本単位 (多角形) を操作するツールです。

領域上で、左クリックすると、その領域が選択されます。

Shift をクリックしながら、領域をクリックすると、その領域も選択状態に加わります。

領域上で、左クリックを押しながらドラッグすると、その領域が移動します。

選択やドラッグといった操作を行うと、右側のインスペクタに情報が表示されます。

領域上で、右クリックすると、コンテキストメニューが表示されます。

削除

選択されている領域が削除されます。

回転

選択されている領域が回転します。

拡大縮小

選択されている領域が拡大縮小します。

グループ化

選択されている領域がグループ化します。

グループ解除

選択されている領域をグループ解除します。

背面に移動

選択されている領域を背面に移動します。

最背面に移動

選択されている領域を最背面に移動します。

前面に移動

選択されている領域を前面に移動します。

最前面に移動

選択されている領域を最前面に移動します。

13.8.2 矩形作成ツール

矩形作成ツールは、矩形の領域を作成するツールです。

このツールを選択すると、編集モードタブの下部に、「正の領域」と「負の領域」が表示されます。

地図エディタタブにてドラッグする事で、選択している種類の矩形領域が作成されます。

13.8.3 多角形作成ツール

多角形作成ツールは、多角形の領域を作成するツールです。

編集モードタブの下部に、「正の領域」と「負の領域」があります。

地図エディタタブにてクリックすると、最初の多角形の頂点が作成されます。続いてクリックする度に、多角形の頂点が追加されていきます。辺が交差するような多角形は作成できません。右クリックすると最後の点が削除されます。3 点以上頂点が作成されている状態で最初の頂点の丸にマウスを移動すると、領域が赤く表示され、そこでクリックすると、多角形が確定されます。

注意: 多角形選択中はメニュー操作等ができなくなります。多角形作成をキャンセルしたい場合、右クリックを連続して押して下さい。

13.8.4 閉領域作成ツール

閉領域作成ツールは、背景の DXF 情報に基づいて閉領域を作成するツールです。

このツールは、背景に DXF が設定されている時のみに選択可能です。背景に DXF が設定されていない時は選択出来ません。

背景に DXF が設定されている場合、背景に DXF のエッジが表示されています。その DXF 上で任意の点をクリックすると、その領域が含まれる閉領域が多角形領域として認識されます。

領域の認識は DXF のバージョンによっては誤動作する場合があります。認識可能な DXF エンティティは、LINE, LWPOLYLINE, CIRCLE, ARC のみです。CIRCLE, ARC に関しては、読み込み時に LINE に分割されます。認識方法としては、クリック点から下方に垂線を降ろし、その点から反時計回りに、一周した領域を認識します。領域内部に領域があった場合でも、それらも含めて、領域として認識します。微小な間隔のノード群は同一点と判定します。

オリジナルの DXF ファイルに領域認識に必要な情報がある場合でも、それらの線分も境界と判断してしまいます。その場合、DXF を編集する外部ツール等を利用して、必要のないレイヤーを削除するなどの前処理を行う事を推奨します。あるいは、背景情報はあくまでも参考として、その上に矩形作成ツール、多角形作成ツールを利用して、領域を作成していく方法もあります。

13.8.5 属性管理テーブル

ユーザー定義領域はレイヤーごとに属性を持ちます。属性は「属性名」と「属性値」のペアの集合として表現されます。

編集モードのレイヤー名で右クリックすると、コンテキストメニューが表示され、属性管理テーブルを表示できます。

属性管理テーブルでは、属性を追加したり、属性を右クリックしてコンテキストメニューから属性を操作できます。

属性はインスペクタに表示されます。

13.9 Undo/Redo

全ての操作は、Undo/Redo 出来ます。

13.10 ファイルメニュー

13.10.1 地図初期化

地図初期化を選択すると、「横 (m)」、「縦 (m)」、「空間参照系」、「画面中心経度」、「画面中心緯度」を入力するダイアログが表示されます。

適切な値を入力し、OK すると、地図が指定されたサイズに初期化されます。「空間参照系」、「画面中心経度」、「画面中心緯度」は後述の GeoJSON インポート／エクスポート以外の操作では利用しません。

13.10.2 地図読み込み

SFM 地図エディタで作成された地図情報 (拡張子 svmap) を読み込みます。

地図ファイルは、プロジェクトフォルダ内に、拡張子 svmap という名前で data フォルダ内に保存されています。

13.10.3 背景 DXF 読み込み

背景に DXF ファイルを読み込みます。

DXF のバージョンによっては誤動作する場合があります。認識可能な DXF エンティティは、LINE, LWPOLYLINE, CIRCLE, ARC のみです。CIRCLE, ARC に関しては、読み込み時に LINE に分割されます。

地図空間上のどこに、DXF の図形情報を配置するかを指定します (図 13.6)。

上に表示されている DXF の図形をドラッグする事で、配置場所を移動出来ます。

地図自体のサイズを変更したい場合、適切なサイズに地図を初期化してから、DXF を配置して下さい。

DXF を配置すると、例えば、図 13.7 のような画面になります。

認識された線分が表示されています。この時点では、背景に DXF が設定されているだけで、領域としては認識されていません。領域操作モードにて移動可能領域を設定する必要があります。なお、DXF を読み込んだ場合、領域操作モードにて、閉領域作成ツールが利用可能になっています。

13.10.4 背景 PNG 読み込み

背景に png ファイルを読み込みます。

背景の縮尺は、地図のサイズに合わせて、変更されます。

地図自体のサイズを変更したい場合、適切なサイズに地図を初期化してから、png を配置して下さい。

png を配置すると、例えば、図 13.8 のような画面になります。

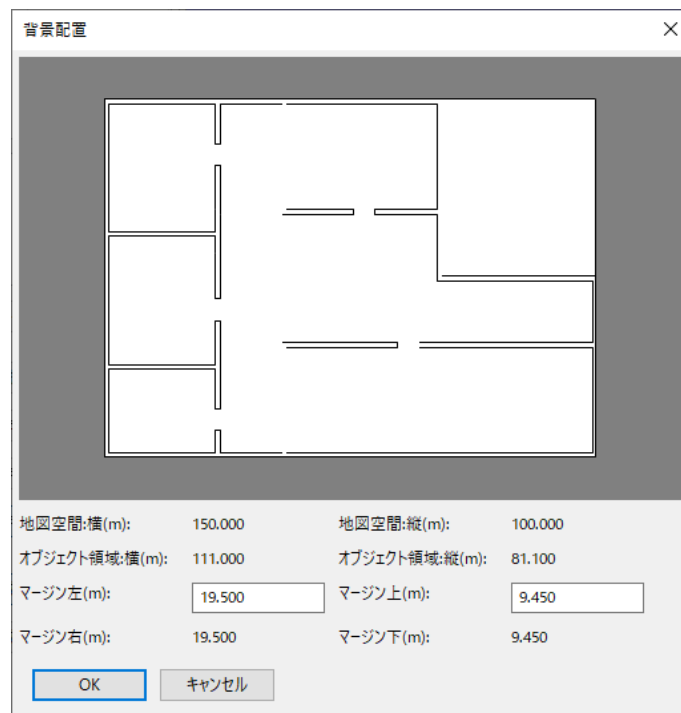


図 13.6: DXf ファイル読み込み

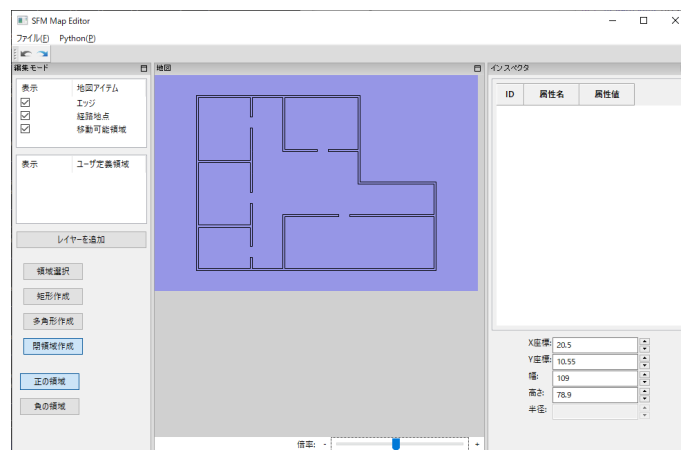


図 13.7: DXF 読み込み後

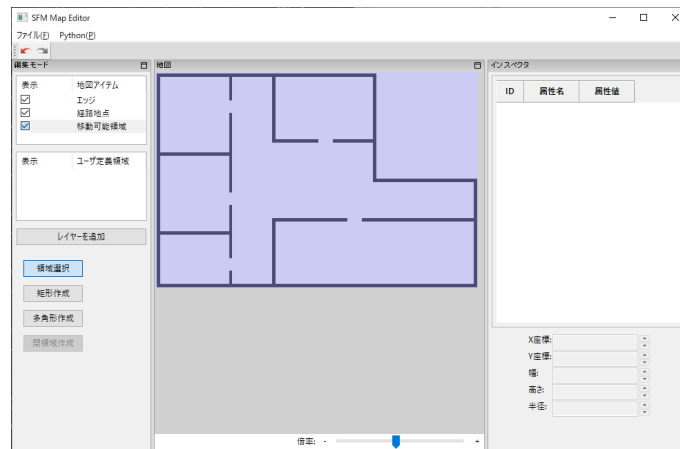


図 13.8: PNG 読み込み後

背景に png が設定されましたが、png はあくまでも領域作成のガイドになるものです。背景画像を基に、領域操作モードにて移動可能領域を設定する必要があります。なお、領域操作モードの、閉領域作成ツールは利用出来ません。

13.10.5 GeoJSON インポート

GeoJSON ファイルを読み込みます。

GeoJSON ファイルの空間参照系は CRS84 (EPSG:4326) であり、トップレベルは FeatureCollection であり、図形はすべて多角形 (Polygon か Multi-Polygon) である必要があります。

移動可能領域、またはユーザー定義領域のどちらかを指定して読み込みます。

GeoJSON ファイルを読み込むと、例えば、図 13.9 のような画面になります。

GeoJSON インポートでは図形をそのままの位置と大きさと読み込むため、必要に応じて地図エディタの縦横が拡大されます。

13.10.6 GeoJSON エクスポート

移動可能領域またはレイヤーを GeoJSON ファイルとしてエクスポートします。

13.10.7 地図設定

地図設定を選択すると、「横 (m)」、「縦 (m)」、「空間参照系」、「画面中心経度」、「画面中心緯度」を入力するダイアログが表示されます。

13.11 Python メニュー



図 13.9: GeoJSON 読み込み後

適切な値を入力し、OK すると、地図の設定が更新されます。

13.10.8 地図保存

作成した地図情報を保存します。

13.11 Python メニュー

13.11.1 Python コンソールを開く

Python コンソールを開きます。ここで地図エディタ用の API を実行することで、プログラマ的に地図エディタを操作できます。

APIの詳細については psim 言語リファレンスマニュアルを参照して下さい。

第14章 強化学習モデル

モデル上に配置する強化学習モデルについて説明します。

強化学習モデルは、強化学習によるセクタ 7.2.7 章や同期エージェント 11.1 章 で使用する強化学習モデルを表します。

14.1 強化学習モデルの編集

強化学習モデルの下のラベルの以外の場所を左ダブルクリック、あるいは、強化学習モデルの配置されている場所を右クリックして表示されるメニュー (3.5.1 章) から「編集」を選ぶと、強化学習モデルの編集画面が表示されます。

編集画面は、以下のタブで構成されます。

14.1.1 共通設定タブ

共通設定タブは 図 14.1 のような画面構成になります。

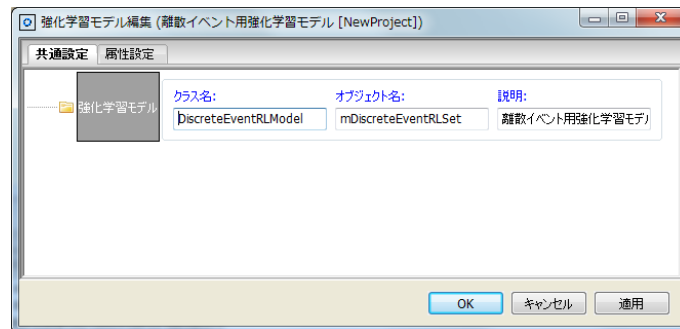


図 14.1: 強化学習モデル共通設定タブ

全ての強化学習モデルで共通の属性を編集できます。

クラス名

自動生成コード (5.8.1 章) におけるこの強化学習モデルのクラス名になります。通常は編集する必要はありません。

オブジェクト名

14.2 離散イベント用強化学習モデル (DiscreteEventRLModel)

自動生成コード (5.8.1 章) におけるこの強化学習モデルのインスタンス名になります。通常は編集する必要はありません。

説明

モデル編集画面におけるこの強化学習モデルの名前になります。5.3.15 章の方法でも変更する事が出来ます。

14.1.2 属性設定タブ

属性設定タブは 図 14.2 のような画面構成になります。

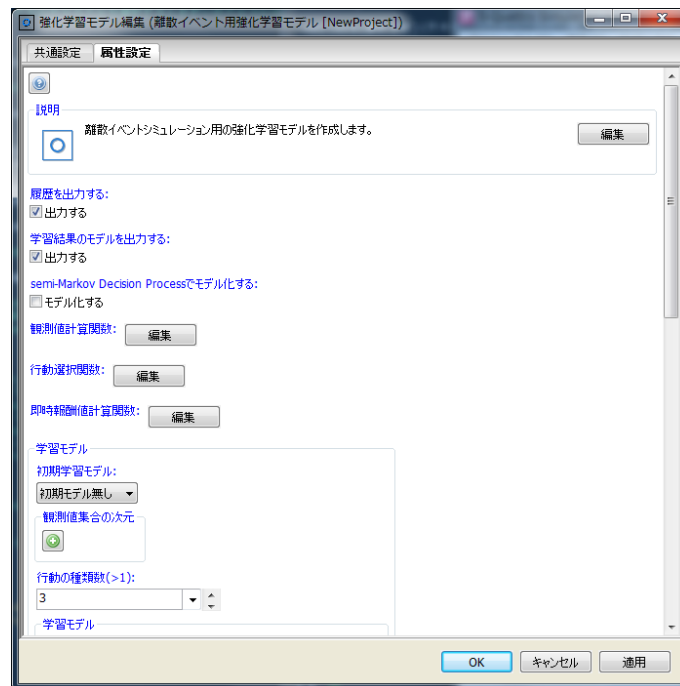


図 14.2: 強化学習モデル属性設定タブ

上のヘルプボタンを押すと、その強化学習モデルの説明が表示されます。強化学習モデルの種類ごとに表示される属性は変わります。各強化学習モデルの振る舞いを定めます。



図 14.3: 離散イベント用強化学習モデル

14.2 離散イベント用強化学習モデル (DiscreteEventRLModel)

離散イベントシミュレーション用の強化学習モデルを作成します。「離散イベント用強化学習モデル」(図 14.3) は離散イベントシミュレーションにおけるアイテムの振り分けに使用される (7.2.7 章) 強化学習モデルを定義します。

履歴を出力する

チェックを入れると、シミュレーション終了後に強化学習の観測値・行動値などの履歴をまとめたモニタを出力します。モニタは「アイコン名 - 履歴」という名前で出力されます。

学習結果のモデルを出力する

チェックを入れると、シミュレーション終了後に強化学習の学習結果を出力します。学習結果は「アイコン名 - 学習モデル」という名前で出力されます。

semi-Markov Decision Process でモデル化する

チェックを入れると、semi-Makov Decision Process によりモデル化され、学習が行われます。

観測値計算関数

観測値を計算する関数を定義します。

行動選択関数

強化学習によって提案された行動に問題がある場合に、この関数で修正します。

即時報酬値計算関数

即時報酬値を計算する関数を定義します。

初期学習モデル

初期学習モデルとして入力フォルダにある強化学習の学習結果のモデルが指定できます。「初期モデル無し」は、初期モデルを使用しない場合に指定します。

モデルを指定すると、指定された学習モデルに合わせるため、以降の設定項目の設定内容や選択肢が変化することがあります。

観測値集合の次元

観測値集合の各次元を設定します。整数または実数値の閉区間か、文字列の有限集合の積集合として観測値集合はあらわされます。ここでは、そのような観測値集合の各次元を設定します。「+」を押すと末尾に次元が追加され、「-」を押すと次元が除去されます。各次元の設定項目には次があります。

名前

次元の名称です。英数字からなる文字列を指定して下さい。

型

次元の値の型です。整数、実数、文字列のいずれかを選択できます。

最小値

型に整数または実数を選択した場合に指定できます。取りうる値の最小値です。

最大値

型に整数または実数を選択した場合に指定できます。取りうる値の最大値です。

取りうる値

型に文字列を選択した場合に指定できます。取りうる値を英数字で指定して下さい。

行動の種類数

行動の種類数です。離散イベント用強化学習モデルは、離散イベントシミュレーションにおけるフローアイテムの振り分け用の強化学習モデルを想定しているため、行動集合は振り分け型に対応する 1 次元の離散有限集合であり、この設定項目は集合の大きさ (= 行動の種類数) をあらわします。

学習モデル

学習手法などに関する設定を行います。詳しくは [14.4 章](#) を参照して下さい。

14.3 エージェント用強化学習モデル (AgentRLModel)



図 14.4: エージェント用強化学習モデル

エージェントシミュレーション用の強化学習モデルを作成します。「エージェント用強化学習モデル」(図 14.4) はエージェントの行動決定に使用される ([7.2.7 章](#)) 強化学習モデルを定義します。

履歴を出力する

チェックを入れると、シミュレーション終了後に強化学習の観測値・行動値などの履歴をまとめたモニタを出力します。モニタは「アイコン名 - 履歴」という名前で出力されます。

学習結果のモデルを出力する

チェックを入れると、シミュレーション終了後に強化学習の学習結果を出力します。学習結果は「アイコン名 - 学習モデル」という名前で出力されます。

semi-Markov Decision Process でモデル化する

チェックを入れると、semi-Makov Decision Process によりモデル化され、学習が行われます。

観測値計算関数

観測値を計算する関数を定義します。

行動選択関数

強化学習によって提案された行動に問題がある場合に、この関数で修正します。

即時報酬値計算関数

即時報酬値を計算する関数を定義します。

初期学習モデル

初期学習モデルとして入力フォルダにある強化学習の学習結果のモデルが指定できます。「初期モデル無し」は、初期モデルを使用しない場合に指定します。

モデルを指定すると、指定された学習モデルに合わせるため、以降の設定項目の設定内容や選択肢が変化することがあります。

観測値集合の次元

観測値集合の各次元を設定します。整数または実数値の閉区間か、文字列の有限集合の積集合として観測値集合はあらわされます。ここでは、そのような観測値集合の各次元を設定します。「+」を押すと末尾に次元が追加され、「-」を押すと次元が除去されます。各次元の設定項目には次があります。

名前

次元の名称です。英数字からなる文字列を指定して下さい。

型

次元の値の型です。整数、実数、文字列のいずれかを選択できます。

最小値

型に整数または実数を選択した場合に指定できます。取りうる値の最小値です。

最大値

型に整数または実数を選択した場合に指定できます。
取りうる値の最大値です。

取りうる値

型に文字列を選択した場合に指定できます。取りうる値を英数字で指定して下さい。

行動値集合の次元

行動値集合の各次元を設定します。整数または実数値の閉区間か、文字列の有限集合の積集合として観測値集合はあらわされます。ここでは、そのような観測値集合の各次元を設定します。「+」を押すと末尾に次元が追加され、「-」を押すと次元が除去されます。各次元の設定項目には次があります。

名前

次元の名称です。英数字からなる文字列を指定して下さい。

型

次元の値の型です。整数、実数、文字列のいずれかを選択できます。

最小値

型に整数または実数を選択した場合に指定できます。
取りうる値の最小値です。

最大値

型に整数または実数を選択した場合に指定できます。
取りうる値の最大値です。

取りうる値

型に文字列を選択した場合に指定できます。取りうる値を英数字で指定して下さい。

学習モデル

学習手法などに関する設定を行います。詳しくは [14.4 章](#) を参照して下さい。

14.4 強化学習モデルの学習モデル

各強化学習モデルの学習モデルの設定について述べます。S⁴ Simulation System の強化学習モデルでは大きく分けて次の学習モデルが使用できます。

行動価値関数を使用したモデル 行動価値関数を用いたモデルです。

ActorCritic モデル Critic と呼ばれる状態価値関数と、行動を決める Actor から構成されるモデルです。

上記の 2 つのモデルそれぞれで設定項目は異なります。それぞれ順に説明します。

14.4.1 行動価値関数を使用したモデル

行動価値関数を使用したモデルでは次のような設定項目があります。

即時報酬の割引率

即時報酬の割引率を設定します。0 以上 1 以下の実数値です。

行動価値関数の形式

行動価値関数の形式を指定します。形式はそれぞれ以下から選択できます。選択した形式によっては追加の設定項目が必要になります。

表形式

近似表現を用いずに価値を明示的に保持する行動価値関数です。以下の設定項目があります。

初期値

値は全てこの値で初期化されます。

NeuralNet

(Feed-Forward) Neural Network によって 行動価値関数 を表現します。以下の設定項目があります。

隠れ層のニューロン数

隠れ層のニューロン数を指定します。各層のニューロン数をカンマで区切って入力します。例えば、隠れ層が 1 枚で、隠れ層のニューロン数が 3 の場合は「3」と入力します。隠れ層が 2 枚で、それぞれのニューロン数が（入力に近い層から）3、4 個の場合は、「3,4」と入力します。

初期重みの最小値

初期重みの最小値です。重みはこの値以上、初期重みの最大値以下の値で初期化されます。

初期重みの最大値

初期重みの最大値です。重みは初期重みの最小値以上、この値以下の値で初期化されます。

HashTiling

(UNH-CMAC) Hash-Tiling [4] によって行動価値関数を表現します。以下の設定項目があります。

タイル数

1 度の計算でアクティブになるタイルの数です。正の整数が指定できます。

メモリーサイズ

タイルの総数です。正の整数が指定できます。

方策

行動価値関数から行動を決める際の方法を指定します。以下の設定項目があります。

方式

行動を決める方法です。以下から選択できます。

ϵ -greedy

ϵ -greedy と呼ばれる方式で行動を選択します。つまり、確率 ϵ でランダムに行動を選択し、確率 $1 - \epsilon$ で価値が最大の行動を選択します。

softmax

行動価値関数から定まる Softmax な分布に従って行動を選択します。Softmax な分布とは、状態 s における行動 a をとる確率が次のようになる分布です。

$$\frac{1}{Z} \frac{Q(s, a)}{\tau}$$

ここで、 Z は正規化定数であり、 Q は行動価値関数、 $\tau > 0$ は温度とよばれるパラメータです。

ϵ または温度

ϵ または温度を表します。

学習手法

学習の手法（行動価値関数の更新手法）を設定します。「学習を行わない」を選択すると、学習は行われません（行動価値関数は更新されません）。それ以外の手法は次の通りです。各手法の選択に伴う設定項目についても合わせて述べます。

Q 学習

Q 学習によって学習します。以下の設定項目を持ちます。

学習率

学習率です。0 以上 1 以下の実数値が指定できます。

TD(λ)

TD(λ) によって学習します。以下の設定項目を持ちます。

学習率

学習率です。0 以上 1 以下の実数値が指定できます。

トレース

トレースに関する設定です。

λ (減衰率)

トレースの減衰率です。0 以上 1 以下の実数値が指定できます。

トレースの種別

トレースの種別です。Accumulate と Replacing の 2 種類から選択できます。

Monte Carlo

Monte Carlo 法によって学習します。以下の設定項目を持ちます。

first-visit または every-visit

first-visit Monte Carlo か every-visit Monte Carlo を指定します。チェックを入れると every-visit で、入れないと first-visit です。

GQ(λ)

GQ(λ) [3] によって学習します。以下の設定項目を持ちます。

学習率

学習率です。0 以上 1 以下の実数値が指定できます。

トレース

トレースに関する設定です。

λ (減衰率)

トレースの減衰率です。0 以上 1 以下の実数値が指定できます。

トレースの種別

トレースの種別です。Accumulate と Replacing の 2 種類から選択できます。

relative step size

relative step size です。0 以上 1 以下の値を設定できます。

14.4.2 ActorCritic モデル

ActorCritic モデルでは次のような設定項目があります。

状態価値関数の形式

状態価値関数の形式を指定します。形式はそれぞれ以下から選択できます。選択した形式によっては追加の設定項目が必要になります。

表形式

近似表現を用いずに価値を明示的に保持する状態価値関数です。以下の設定項目があります。

初期値

値は全てこの値で初期化されます。

NeuralNet

(Feed-Forward) Neural Network によって 状態価値関数 を表現します。以下の設定項目があります。

隠れ層のニューロン数

隠れ層のニューロン数を指定します。各層のニューロン数をカンマで区切って入力します。例えば、隠れ層が 1 枚で、隠れ層のニューロン数が 3 の場合は「3」と入力します。隠れ層が 2 枚で、それぞれのニューロン数が（入力に近い層から）3、4 個の場合は、「3,4」と入力します。

初期重みの最小値

初期重みの最小値です。重みはこの値以上、初期重みの最大値以下の値で初期化されます。

初期重みの最大値

初期重みの最大値です。重みは初期重みの最小値以上、この値以下の値で初期化されます。

HashTiling

(UNH-CMAC) Hash-Tiling [4] によって状態価値関数を表現します。以下の設定項目があります。

タイル数

1 度の計算でアクティブになるタイルの数です。正の整数が指定できます。

メモリーサイズ

タイルの総数です。正の整数が指定できます。

方策・Actor

行動の決め方をあらわす方策・Actor を指定します。指定できる方策には以下の種別があり、それぞれで設定内容も異なります。順に述べていきます。

Epsilon-Greedy

ϵ -greedy と呼ばれる方式で行動を選択します。以下の設定項目を含みます。

Preference の形式

Preference の形式を指定します。形式はそれぞれ以下から選択できます。選択した形式によっては追加の設定項目が必要になります。

表形式

近似表現を用いずに価値を明示的に保持する Preference です。以下の設定項目があります。

初期値

値は全てこの値で初期化されます。

NeuralNet

(Feed-Forward) Neural Network によって Preference を表現します。以下の設定項目があります。

隠れ層のニューロン数

隠れ層のニューロン数を指定します。各層のニューロン数をカンマで区切って入力します。例えば、隠れ層が 1 枚で、隠れ層のニューロン数が 3 の場合は「3」と入力します。隠れ層が 2 枚で、それぞれのニューロン数が（入力に近い層から）3、4 個の場合は、「3,4」と入力します。

初期重みの最小値

初期重みの最小値です。重みはこの値以上、初期重みの最大値以下の値で初期化されます。

初期重みの最大値

初期重みの最大値です。重みは初期重みの最小値以上、この値以下の値で初期化されます。

HashTiling

(UNH-CMAC) Hash-Tiling [4] によって Preference を表現します。以下の設定項目があります。

タイル数

1 度の計算でアクティブになるタイルの数です。正の整数が指定できます。

メモリーサイズ

タイルの総数です。正の整数が指定できます。

ϵ

ϵ です。

Softmax

Preference から定まる Softmax な分布に従って行動を選択します。以下の設定項目を含みます。

Preference の形式

Preference の形式を指定します。形式はそれぞれ以下から選択できます。選択した形式によっては追加の設定項目が必要になります。

表形式

近似表現を用いずに価値を明示的に保持する Preference です。以下の設定項目があります。

初期値

値は全てこの値で初期化されます。

NeuralNet

(Feed-Forward) Neural Network によって Preference を表現します。以下の設定項目があります。

隠れ層のニューロン数

隠れ層のニューロン数を指定します。各層のニューロン数をカンマで区切って入力します。例えば、隠れ層が 1 枚で、隠れ層のニューロン数が 3 の場合は「3」と入力します。隠れ層が 2 枚で、それぞれのニューロン数が（入力に近い層から）3、4 個の場合は、「3,4」と入力します。

初期重みの最小値

初期重みの最小値です。重みはこの値以上、初期重みの最大値以下の値で初期化されます。

初期重みの最大値

初期重みの最大値です。重みは初期重みの最小値以上、この値以下の値で初期化されます。

HashTiling

(UNH-CMAC) Hash-Tiling [4] によって Preference を表現します。以下の設定項目があります。

タイル数

1 度の計算でアクティブになるタイルの数です。正の整数が指定できます。

メモリーサイズ

タイルの総数です。正の整数が指定できます。

温度

Softmax 分布における温度です。

Gaussian (Neural Net)

ガウス分布 $\pi(s, a) = \frac{1}{Z} \exp((a - \mu(s))^T \Sigma^{-1} (a - \mu(s)))$ によって行動が選択される方策です。ただし、平均 $\mu(s)$ は (フィードフォワード) ニューラルネットにより定まります。整数値次元に対しては、ガウス分布からサンプリングした結果の値を最も近い整数に丸めた値が使用されます。ガウス分布からサンプリングした結果、連続値・整数値の次元において範囲外の値が得られた場合は、範囲内の値に丸められます。行動値集合の次元に文字列が含まれる場合は使えません。

以下の設定項目を含みます。

隠れ層のニューロン数

隠れ層のニューロン数を指定します。各層のニューロン数をカンマで区切って入力します。例えば、隠れ層が 1 枚で、隠れ層のニューロン数が 3 の場合は「3」と入力します。隠れ層が 2 枚で、それぞれのニューロン数が（入力に近い層から）3、4 個の場合は、「3,4」と入力します。

初期重みの最小値

初期重みの最小値です。重みはこの値以上、初期重みの最大値以下の値で初期化されます。

初期重みの最大値

初期重みの最大値です。重みは初期重みの最小値以上、この値以下の値で初期化されます。

Gaussian (Hash Tiling)

ガウス分布 $\pi(s, a) = \frac{1}{2} \exp((a - \mu(s))^T \Sigma^{-1} (a - \mu(s)))$ によって行動が選択される方策です。ただし、平均 $\mu(s)$ は特徴ベクトル $\phi(s)$ と行列 T により、 $\mu(s) = T\phi(s)$ とあらわせるものとします。ここで $\phi(s)$ は Hash Tiling によって得られるベクトルとします。その他の性質は Gaussian (Neural Net) と同様です。

以下の設定項目を含みます。

タイル数

1 度の計算でアクティブになるタイルの数です。正の整数が指定できます。

メモリーサイズ

タイルの総数です。正の整数が指定できます。

学習手法

学習の手法（状態価値関数と方策・Actor の更新手法）を設定します。「学習を行わない」を選択すると、学習は行われません（状態価値関数と方策・Actor は更新されません）。それ以外の手

法は次の通りです。各手法の選択に伴う設定項目についても合わせて述べます。

Actor Critic

Actor Critic に従って学習します。以下の設定項目を含みます。

Critic の手法

状態価値関数の更新手法の設定です。以下の更新手法から選択できます。手法ごとに追加の設定項目があります。

TD(λ) Critic

TD(λ) に従って状態価値関数を更新します。以下の設定項目を含みます。

即時報酬の割引率

即時報酬の割引率を設定します。0 以上 1 以下の実数値です。

学習率

学習率です。0 以上 1 以下の実数値が指定できます。

トレース

トレースに関する設定です。

λ (減衰率)

トレースの減衰率です。0 以上 1 以下の実数値が指定できます。

トレースの種別

トレースの種別です。Accumulate と Replacing の 2 種類から選択できます。

Average Reward TD(λ) Critic

Average Reward TD(λ) に従って状態価値関数を更新します。以下の設定項目を含みます。

学習率

学習率です。0 以上 1 以下の実数値が指定できます。

トレース

トレースに関する設定です。

λ (減衰率)

トレースの減衰率です。0 以上 1 以下の実数値が指定できます。

トレースの種別

トレースの種別です。Accumulate と Replacing の 2 種類から選択できます。

平均即時報酬推定に関する学習率

平均即時報酬推定に関する学習率です。0 以上 1 以下の実数値が指定できます。

Actor の学習率

Actor の学習率です。0 以上 1 以下の実数値が指定できます。

Natural Actor Critic

Average Reward TD(λ) Critic

Average Reward TD(λ) に従って状態価値関数を更新します。以下の設定項目を含みます。

学習率

学習率です。0 以上 1 以下の実数値が指定できます。

トレース

トレースに関する設定です。

λ (減衰率)

トレースの減衰率です。0 以上 1 以下の実数値が指定できます。

トレースの種別

トレースの種別です。Accumulate と Replacing の 2 種類から選択できます。

平均即時報酬推定に関する学習率

平均即時報酬推定に関する学習率です。0 以上 1 以下の実数値が指定できます。

Actor の学習率

Actor の学習率です。0 以上 1 以下の実数値が指定できます。

advantage step size

advantage step size です。0 以上 1 以下の実数値が指定できます。

第15章 3Dアニメーション

モデル上に配置する3Dアニメーションについて説明します。3Dアニメーションとは、シミュレーションの実行結果を3Dアニメーションとして表示させるビューワを管理するものです。SFM環境オブジェクト、SFM地図オブジェクト、NW地図オブジェクトを用いたシミュレーションを3Dアニメーションとして表示させることができます。

15.1 3Dアニメーション (Animation3D)



図 15.1: 3Dアニメーション

3Dアニメーションを表示します。「環境オブジェクト」は、3Dアニメーションを表示する環境を指定します。SFM環境オブジェクト、SFM地図オブジェクト、NW地図オブジェクトを3Dアニメーションとして表示することができます。

15.1.1 属性設定タブ

アニメーション編集画面を開くと、最初に属性設定タブが開きます。属性設定タブでは、図 15.2 のような画面構成になります。上のヘルプボタンを押すと、その3Dアニメーションの説明が表示されます。

15.13D アニメーション (Animation3D)

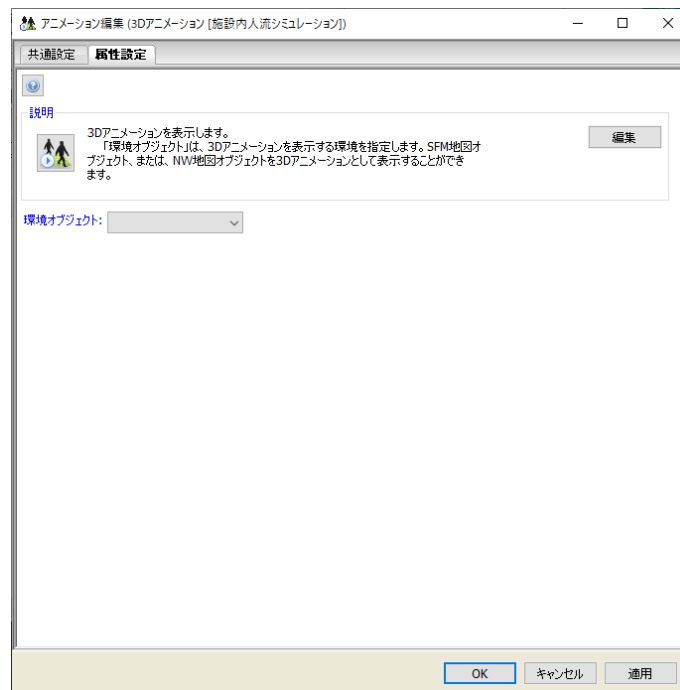


図 15.2: アニメーション属性設定タブ

第16章 可視化ダッシュボード

可視化ダッシュボードについて説明します。可視化ダッシュボードとは、エージェントシミュレーションの結果を可視化する機能であり、ブラウザ上で動作します。SFM 環境オブジェクト、SFM 地図オブジェクト、または NW 地図オブジェクトを利用したシミュレーションを対象としています。

16.1 起動と停止

16.1.1 準備

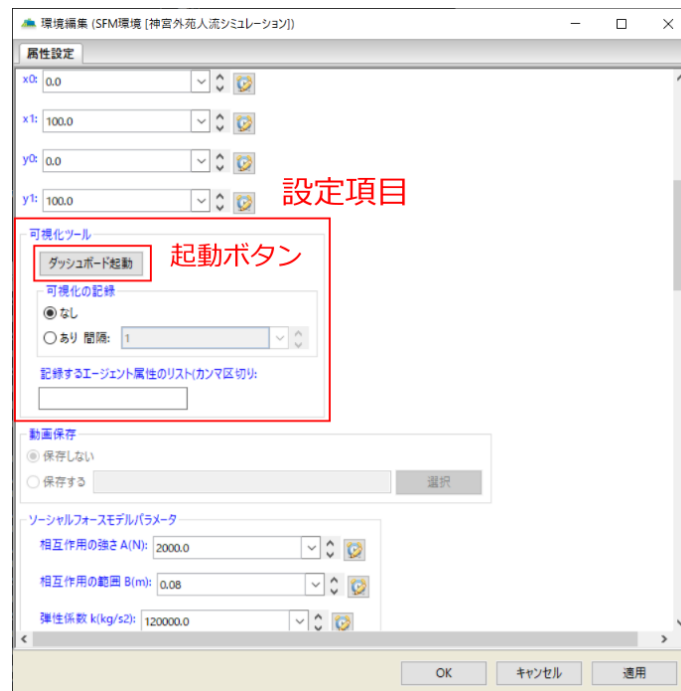


図 16.1: 設定画面

可視化ダッシュボードの設定画面は、SFM 環境オブジェクト、SFM 地図オブジェクト、または NW 地図オブジェクトの編集画面にあります。

可視化の記録を「あり」に設定してシミュレーションを実行すると、プロジェクトの data フォルダに XXX_record というフォルダが生成され、その中に可視化に必要なデータが出力されます。例えば、「間隔」に指定された秒数ごとにエージェントの位置が記録されます。

また「記録するエージェント属性のリスト」に属性（コード編集などを通じてエージェントに定義されたインスタンス変数の名前）を指定すると、その時系列も合わせて出力されます。

16.1.2 起動方法

設定項目の中の起動ボタンを押してください。起動には数秒かかりますので、ボタンをクリックしたらそのままお待ちください。

起動したら、お使いのパソコンで既定のブラウザとして設定されているブラウザが自動的に立ち上がります。同じ URL にアクセスすることで新しいタブもしくは他のブラウザでも作業できますが、アクセスする度にそれまでの作業内容が初期化されます。したがって、1つのブラウザ・1つのタブに限ってご利用ください。

16.1.3 停止方法

起動時に立ち上がるコマンドプロンプトを閉じると、サーバーも合わせて停止します。

16.2 操作方法

可視化ダッシュボードの操作方法について説明します。

16.2.1 構成

可視化ダッシュボードはメニューとグリッドパネルから構成されます。グリッドパネルにコンポーネントを追加していくことで、自由にダッシュボードを作成することができます。

16.2.2 メニュー

メニューにはセーブボタン、ロードボタン、配置設定、追加設定があります。

セーブボタンを押すとダッシュボードの状態を記録した pickle ファイルが保存されます。自動的にダウンロードフォルダに保存されるか、保存先を自

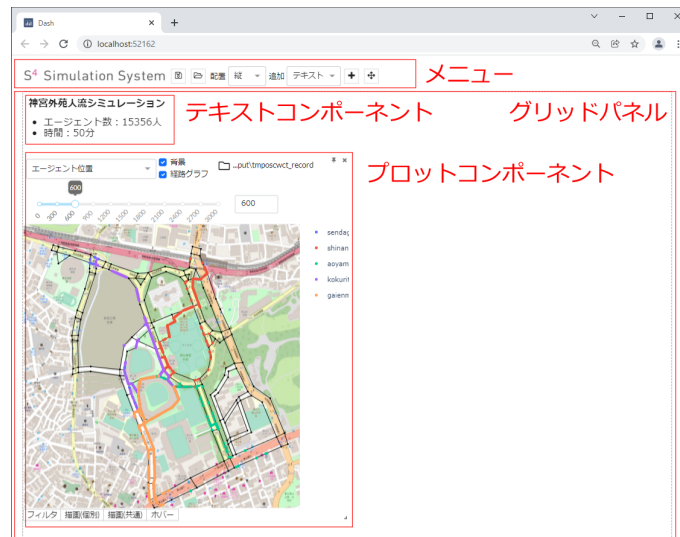


図 16.2: 可視化ダッシュボード

分で決めるダイアログが表示されるかは、お使いのブラウザの設定で変更することができます。

ロードボタンを押すとファイル選択ダイアログが表示され、pickle ファイルを選択すると保存したダッシュボードが読み込まれます。ロードボタンにドラッグ&ドロップして読み込むことも可能です。

配置設定では、コンポーネントを整列するための力を設定することができます。縦のときコンポーネントは上方向に引っ張られ、横のとき左方向に引っ張られます。自由のとき力は働きません。

追加設定は、追加するコンポーネントの種類を選択するドロップダウンと、追加ボタン、ドラッグ要素から成ります。追加ボタンを押すか、ドラッグ要素をグリッドパネルにドラッグ&ドロップすることでコンポーネントを追加することができます。ただしグリッドパネルにコンポーネントが1つもない場合は、追加ボタンでのみ追加可能です。

16.2.3 グリッド

各コンポーネントを乗せたグリッドの操作について説明します。

グリッドの左上または上辺中央にカーソルを乗せるとドラッグ要素が描画されます。ドラッグ要素をつかむことでコンポーネントを移動させることができます。

グリッドの右上にはピンボタンと削除ボタンが描画されています。ピンボタンを押すとグリッドが固定され、位置および中身を変更することができなくなります。ピンボタンをもう一度押すと固定は解除されます。削除ボタン

を押すと確認ダイアログが表示され、OK ボタンを押すとコンポーネントが削除されます。一度削除すると取り消しはできないのでご注意ください。

グリッドの右下には拡大・縮小ハンドルが描画されています。ハンドルをドラッグすることでグリッドのサイズを変更することができます。グリッドの横幅に合わせて、コンポーネントの横幅が決まります。一方でグリッドの高さは、コンポーネントの高さが変わる度に再計算され、自動的に伸縮します。

16.3 プロットコンポーネント

プロットコンポーネントについて説明します。

16.3.1 プロジェクトフォルダの選択

プロットコンポーネントの右上にあるフォルダアイコンをクリックすると、ファイル選択ダイアログが開きます。他プロジェクトの data/XXX_record 内にある path.info.txt を選択すると、プロット内容がそのプロジェクトのものに切り替わります。

なお、path.info.txt には XXX_record までの絶対パスが記入されています。可視化の記録を「あり」に設定してシミュレーションを実行すると自動的にファイルが用意されますが、ユーザーが独自に用意したもので構いません。

16.3.2 プロット種類の変更方法

プロットコンポーネントの上部にあるドロップダウンメニュー (初期値は「エージェントの位置」となっています。) から描画内容を変更します。

16.3.3 時間スライダー

プロットコンポーネントの上部には時間スライダーが表示されます。描画の種類によって片側スライダーか両側スライダーのいずれかが表示されます。

時間スライダーを変更することで指定した時間 (両側の場合は指定した時間範囲) のデータに限定した描画を行うことができます。

指定する時間の値はシミュレーション時に S4 の環境部品の属性設定タブ内の「可視化ツール-可視化の記録-間隔」で指定した値の倍数である必要があります。

16.3.4 背景、経路グラフの表示設定

プロット種類の右にある「背景」「経路グラフ」と書かれたチェックボックスのチェックをつける/外すことで背景 (経路グラフ) の描画の切り替えを行うことができます。

16.3.5 設定タブ

フィルタ

プロットコンポーネントの下部に表示されている「フィルタ」タブをクリックするとフィルタの設定が表示されます。

「追加」ボタンをクリックするとフィルタの設定を行うことができます。フィルタで選択できる条件は一致条件か上下限条件のいずれかです。どちらの場合も、シミュレーション時に S4 の環境部品の属性設定タブ内の「可視化ツール-記録するエージェント属性のリスト (カンマ区切り)」で設定したエージェント属性とデフォルトで記録される属性が指定した値に一致または指定した値の範囲内に存在するかどうかの条件フィルタを作成することができます。

フィルタは複数作成することが可能です。複数のフィルタが作成された場合は全ての条件に合致するデータのみが取得されます。OR 条件には対応していません。

フィルタを削除する場合は削除したいフィルタに対応する削除ボタンをクリックします。

描画 (個別)

プロットコンポーネントの下部に表示されている「描画 (個別)」タブをクリックすると描画個別の設定が表示されます。

設定できるパラメータの内容は個別の描画の説明を参照してください。

描画 (共通)

プロットコンポーネントの下部に表示されている「描画 (共通)」タブをクリックすると全ての描画に共通の設定が表示されます。

「描画領域設定」では描画するグラフの横幅とグラフの右の余白幅の割合を指定することができます。グラフの右余白には描画によってはカラーバーなどが表示されます。

グラフの横幅は 10px 以上の値を指定してください。

右余白幅は 0 以上 0.5 以下の値を指定してください。

「背景描画設定」では背景画像の不透明度を設定できます。0以上1以下の値を指定してください。

「経路地点グラフ描画設定」では経路グラフの表示の設定を行うことができます。

経路地点サイズ、経路エッジ太さは0以上の値を指定してください。表示濃度は0以上1以下の値を指定してください。

ホバー

プロットコンポーネントの下部に表示されている「ホバー」タブをクリックするとホバーの設定が表示されます。

「エージェントホバー」「経路地点ホバー」と書かれたトグルスイッチをクリックするとエージェントのホバー表示、経路地点のホバー表示のON/OFFを切り替えることができます。

「エージェントホバー」「経路地点ホバー」の下に書かれているのはエージェント/経路地点の属性です。属性はシミュレーション時に付与したものが表示されます。

各属性の左横にあるチェックボックスをつける/消すことでその属性をホバー表示するかどうかを決めることができます。

ホバー表示は描画の種類によってはホバー表示をONにしても表示されない場合があります。詳しくは各描画の説明を参照してください。

16.3.6 個別の描画の説明

エージェント位置

指定した時間のエージェントの位置を描画します。

「描画 (個別)」で設定できるパラメータは以下の通りです。

エージェント色 エージェントの表示色を指定したエージェントの属性によって色分けします。

経路地点色 経路地点の表示色を指定した経路地点の属性によって色分けします。

マーカーサイズ エージェントの表示サイズを変更します。0以上の値が有効です。

マーカー色 エージェントの表示色を指定します。エージェント色を指定していない場合に有効です。

表示濃度 エージェントの表示の不透明度を指定します。0以上1以下の値が有効です。

エージェント位置 (ヒートマップ)

指定した範囲の時間のある位置における累積のエージェント人数に関するヒートマップを描画します。

エージェントホバー、経路地点ホバー共に表示されません。

「描画 (個別)」で設定できるパラメータは以下の通りです。

背景分割数 背景を縦横ともに指定した数に分割してヒートマップを表示します。1 以上の値が有効です。

人数カウント最小値 カラースケールの最小値です。1 以上の値が有効です。

カラースケール カラースケールを変更します。

ヒートマップ濃度 ヒートマップの不透明度を指定します。0 以上 1 以下の値が有効です。

エージェント位置 (kde)

指定した範囲の時間のある位置における累積のエージェント人数に関するヒートマップを描画します。ヒートマップとの違いは等高線図であるかどうかです。

エージェントホバー、経路地点ホバー共に表示されません。

「描画 (個別)」で設定できるパラメータは以下の通りです。

背景分割数 背景を縦横ともに指定した数に分割してヒートマップを表示します。1 以上の値が有効です。

人数カウント最小値 カラースケールの最小値です。1 以上の値が有効です。

カラースケール カラースケールを変更します。

ヒートマップ濃度 ヒートマップの不透明度を指定します。0 以上 1 以下の値が有効です。

エージェント軌跡

指定したエージェント ID に該当するエージェントの軌跡を描画します。

「描画 (個別)」で設定できるパラメータは以下の通りです。

エージェント ID 描画するエージェント ID です。複数指定可能。

経路地点混雑度

指定した範囲の時間内で各経路地点に存在したエージェントの数 (累積値) を描画します。数が大きいほど表示される点の大きさが大きくなります。

エージェントホバーは表示されません。

「描画 (個別)」で設定できるパラメータは以下の通りです。

人数カウント最小値 カラースケールの最小値です。1 以上の値が有効です。

人数カウント最大値 カラースケールの最大値です。1 以上の値が有効です。

表示サイズ最小値 表示する点の大きさの最小値です。0 以上の値が有効です。

表示サイズ最大値 表示する点の大きさの最大値です。0 以上の値が有効です。

経路地点の濃度 混雑度の描画色の不透明度を指定します。0 以上 1 以下の値が有効です。

カラースケール カラースケールを変更します。

エージェント流量 (経路地点集約)

kmeans 法によって経路地点を指定した数のクラスタに分割し、指定した時間内でのクラスタ間のエージェントの移動人数の累積値を描画します。各クラスタは同一クラスタの経路地点を全て含むような凸包として表示されます。また、クラスタ間の移動人数が矢印で表示されます。

エージェントホバーは表示されません。

「描画 (個別)」で設定できるパラメータは以下の通りです。

クラスタ数 経路地点をいくつのクラスタに分割するか指定します。1 以上の値が有効です。

クラスタ色 クラスタの色を指定します。

クラスタ色の濃度 クラスタの色の描画色の不透明度を指定します。0 以上 1 以下の値が有効です。

流量エッジのカラースケール 移動人数に関する矢印のカラースケールを変更します。

流量エッジの幅 移動人数に関する矢印の幅を指定します。0.1 以上の値が有効です。

流量エッジのカラースケール最大値 移動人数に関する矢印のカラースケール最大値を指定します。0 以上の値が有効です。

出発/到着地 (グリッド)

背景を指定した数に分割した後、指定した時間内におけるエージェントの出発地から到着地への移動人数を表示します。移動人数が多いほど矢印が太く描画されます。

エージェントホバーは表示されません。

「描画 (個別)」で設定できるパラメータは以下の通りです。

背景分割数 背景を縦横ともに指定した数に分割してヒートマップを表示します。1 以上の値が有効です。

カラースケール最大値 移動人数に関する矢印のカラースケール最大値を指定します。0 以上の値が有効です。

カラースケール 移動人数に関する矢印のカラースケールを変更します。

カラースケール濃度 移動人数に関する矢印の描画色の不透明度を指定します。0 以上 1 以下の値が有効です。

矢印幅最小値 移動人数に関する矢印の幅の最小値を指定します。0.1 以上の値が有効です。

矢印幅最大値 移動人数に関する矢印の幅の最大値を指定します。0.1 以上の値が有効です。

表示流量最小値 ここで指定した値以上の移動人数をもつ矢印のみが描画されます。0 以上の値が有効です。

出発/到着地 (経路地点間)

指定した時間内におけるエージェントの出発地から到着地への移動人数を表示します。移動人数が多いほど矢印が太く描画されます。

エージェントホバーは表示されません。

「描画 (個別)」で設定できるパラメータは以下の通りです。

カラースケール最大値 移動人数に関する矢印のカラースケール最大値を指定します。0 以上の値が有効です。

カラースケール 移動人数に関する矢印のカラースケールを変更します。

カラースケール濃度 移動人数に関する矢印の描画色の不透明度を指定します。0 以上 1 以下の値が有効です。

矢印幅最小値 移動人数に関する矢印の幅の最小値を指定します。0.1 以上の値が有効です。

矢印幅最大値 移動人数に関する矢印の幅の最大値を指定します。0.1 以上の値が有効です。

表示流量最小値 ここで指定した値以上の移動人数をもつ矢印のみが描画されます。0 以上の値が有効です。

出発/到着地 (ヒートマップ)

背景を指定した数に分割した後、指定した時間内におけるエージェントの出発地/到着地の人数に関するヒートマップを表示します。

エージェントホバー、経路地点ホバーは表示されません。

「描画 (個別)」で設定できるパラメータは以下の通りです。

出発地/到着地 出発地を選択すると出発地に関するヒートマップ、到着地を選択すると到着地に関するヒートマップとなります。

背景分割数 背景を縦横ともに指定した数に分割してヒートマップを表示します。1 以上の値が有効です。

カラースケール最大値 ヒートマップのカラースケール最大値を指定します。0 以上の値が有効です。

カラースケール ヒートマップののカラースケールを変更します。

ヒートマップ濃度 ヒートマップの不透明度を指定します。0 以上 1 以下の値が有効です。

16.4 テキストコンポーネント

テキストコンポーネントについて説明します。

16.4.1 編集/表示パネル

編集モードのときマークダウン・数式 (\$または\$\$で囲う) を入力することができます。マークダウンは CommonMark、数式は KaTeX に準ずるものを記述してください。

表示モードのときレンダリングされた結果が表示されます。

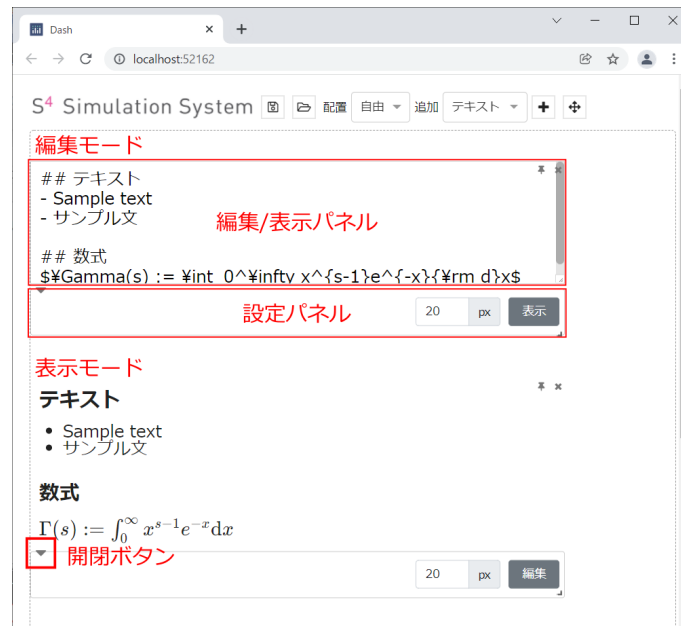


図 16.3: テキストコンポーネント

16.4.2 設定パネル

編集/表示パネル左下の開閉ボタンをクリックすると、設定パネルが開閉します。

設定パネルには文字サイズの入力欄と、表示/編集モードの切り替えボタンがあります。文字サイズの単位はピクセル (px) です。

第17章 コールセンター用部品

コールセンターのシミュレーションに特化した部品です。

17.1 資源

17.1.1 対応待ちリスト (CallQueue)



図 17.1: 対応待ちリスト

「対応待ちリスト」(図 17.1) は、対応待ちリストを表します。

対応待ち上限数

リストに蓄えられる上限数を表します。

記録

対応待ちリストの利用状態を記録するかどうかを表します

17.2 フローアイテム

17.2.1 オペレーター (Operator)



図 17.2: オペレーター

「オペレーター」(図 17.2) は、オペレーターを表します。部品「出勤」(17.3.9章) で生成されます。

出社時間

シミュレーション開始後、入社するまでの時間を指定します。

退社時間

シミュレーション開始後、退社するまでの時間を指定します。

稼働時間

内部で使用します。0 を指定しておいて下さい。

17.2.2 問い合わせ (PhoneCall)



図 17.3: 問い合わせ

「問い合わせ」(図 17.3) は、コールセンターへの問い合わせを表します。部品「入電」(17.3.5 章) で生成されます。

処理時間

この問い合わせの処理時間を示します。

後処理時間

この問い合わせの後処理時間を示します。

問い合わせ ID

この問い合わせの ID が内部的に設定されます。変更しないで下さい。

発生時間

この問い合わせの発生時間が内部的に設定されます。変更しないで下さい。

対応開始時間

この問い合わせの対応開始時間が内部的に設定されます。変更しないで下さい。

対応終了時間

この問い合わせの対応終了時間が内部的に設定されます。変更しないで下さい。

17.3 部品

17.3.1 放棄呼 (Abandon)



図 17.4: 放棄呼

「放棄呼」(図 17.4) は、放棄呼を表します。

17.3.2 時間確認 (CheckTime)



図 17.5: 時間確認

「時間確認」(図 17.5) は、オペレーターが時間を確認します。例えば退社するかどうかなどの確認して条件分岐する場合に使います。

条件式

勤務を続ける条件と、その時の出力ポートを指定します。

それ以外

退社時の出力ポートを指定します。

17.3.3 対応終了 (EndOfOperation)



図 17.6: 対応終了

「対応終了」(図 17.6) は、問い合わせの対応が終了した事を示します。問い合わせの終了時間とオペレーターの稼動時間を設定します。

モニター「応答数」を記録します。列には累積回数が含まれます。モニター「問い合わせの記録」を記録します。列には、問い合わせ ID、発生時間、対応開始時間、対応終了時間が含まれます。

17.3.4 受電 (GetFromCallQueue)



図 17.7: 受電

「受電」(図 17.7) は、問い合わせリストから受電します。

対応待ちリスト

対応待ちリストを指定します。

受電の記録

受電の待ち時間が記録されます。

17.3.5 入電 (GetPhoneCall)



図 17.8: 入電

「入電」(図 17.8) は、入電を表します。

問い合わせ名

問い合わせ種別を指定します。

受け付け開始時間

シミュレーション開始後、受け付けを開始するまでの時間を指定します。

入電間隔

入電の間隔を指定します。



図 17.9: 対応

17.3.6 対応 (Operate)

「対応」(図 17.9) は、オペレーターが問い合わせに対して対応します。

処理時間

問い合わせの対応時間を指定します。

17.3.7 対応待ち (PutToCallQueue)



図 17.10: 対応待ち

「対応待ち」(図 17.10) は、問い合わせを対応待ちリストに追加します。放棄呼になった場合のみ、出力ポートに出力されます。

対応待ちリスト

追加する対応待ちリストを指定します。

放棄呼になる時間

放棄呼になるまでの時間を指定します。

対応待ちの記録

追加結果と追加待ち時間が記録されます。追加結果には、追加/放棄のいずれかが記録されます。

17.3.8 対応開始 (StartOfOperation)

「対応開始」(図 17.11) は、問い合わせの対応の開始を示します。内部で問い合わせの開始時間を設定します。



図 17.11: 対応開始



図 17.12: 出社

17.3.9 出社 (StartWorking)

「出社」(図 17.12) は、オペレーターを出社させます。

オペレーター名

オペレーターを指定します。

出勤人数

そのオペレーターの人数を指定します。

17.3.10 退社 (StopWorking)



図 17.13: 退社

「退社」(図 17.13) は、オペレーターを退社させます。

第18章 窓口業務用部品

窓口業務シミュレーションに特化した部品です。

18.1 資源

18.1.1 窓口 (CounterService)



図 18.1: 窓口

窓口(図 18.1)は、商業施設等の受付窓口を模したファシリティオブジェクトです。ベースとなっているアイコンは、資源「ファシリティ」(8.3章)です。ファシリティに関する部品としては、部品「ファシリティ要求」(6.6章)、部品「ファシリティ解放」(6.7章)、部品「ファシリティ利用」(6.8章)などがあります。

窓口スケジュール

利用可能な窓口のスケジュールを表します。スケジュール無しの場合は、常に窓口にいる状態となり、お客様の対応をします。スケジュール有りでは表指定を選択した場合は、時間カラムおよび1:開,0:閉カラムに指定した値に応じて窓口の人員配置のスケジュールリングができます。時間カラムには、以下の3つの形式で設定が可能です。1. 数値指定たとえば、時間カラムに10と入力すれば、10時からの窓口の開閉状態を変更できます。2. 時刻指定たとえば、07:00:00と指定すれば、7時からの窓口の開閉状態を変更できます。3. 日付時刻指定たとえば、2016-03-09 07:00:00と指定すれば、2016年3月9日7時から窓口の開閉状態を変更できます。

1:開,0:閉カラムには0または1の値のみ設定することができます。1は窓口が開いている状態、0は窓口に係員がおらず、閉まっている状態を示します。

18.2 フローアイテム

18.2.1 お客 (CounterServiceCustomer)



図 18.2: お客

「お客」(図 18.2) は、窓口を利用するお客を表します。部品「窓口到着」(18.3.1 章) で生成されます。

お客 ID

内部で使用します。

来店時刻

内部で使用します。

退店時刻

内部で使用します。

対応開始時間

内部で使用します。

対応終了時間

内部で使用します。

サービス時間

内部で使用します。



図 18.3: 窓口到着

18.3 部品

18.3.1 窓口到着 (EnterCounterService)

部品「窓口到着」(図 18.3) はお客を窓口に着させます。

お客名

この「窓口到着」部品が生成するお客アイテムを表します。

来店開始時間

最初にお客アイテムを生成するのに要する時間を表します。来店間隔の生成方式で、「指数分布 (パラメータ変化)」を指定した場合は、時間及び平均の列カラムを持った入力ファイルを必要とします。入力ファイルで、時間列を数値で指定している場合は、入力ファイルの最初の行で指定した時間と、来店開始時間で指定した時間が一致していなければなりません。入力ファイルの時間列を日時で指定している場合は、来店開始時間を 0 に設定してください。

来店間隔

お客アイテムを生成するのに要する時間を表します。指数分布 (パラメータ変化) を指定した場合は、時間及び平均の列カラムを持った入力ファイルを必要とします。時間列には数値または年月日を指定でき、平均列には 0 より大きい正の値を指定します。平均列の単位は、「平均列の時間の単位」から指定できます。日付のフォーマットは、以下の 2 通りの方法で指定できます。1. HH:MM:SS (H は時間、M は分、S は秒を表します。たとえば、07:30:20 や、20:03:03 など。) この場合は、日付をまたぐシミュレーションは行えません。2. YYYY-MM-DD HH:mm:SS (Y は年、M は月、D は日、H は時間、m は分、S は秒を表します。たとえば、2016-02-03 09:30:30 など。) この場合は、日付をまたぐシミュレーションを行えます。



図 18.4: 対応終了

18.3.2 対応終了 (LeaveCounterService)

部品「対応終了」(図 18.4) はフローの終端です。退店していくお客アイテムに設定された属性リストを monitor に出力します。

18.3.3 窓口選択部品:レジ (SelectCounterServiceMultipleLines)



図 18.5: 窓口選択部品 : レジ

部品「窓口選択部品:レジ」(図 18.5) は、入力されたお客アイテムの出力先を分岐させます。お客アイテムは、デフォルトでは接続先の窓口のうち、最もお客が並んでいない窓口に並びます(最短キュー)。この挙動を変えたい場合、画面上部の共通設定タブを開き、出力ポートの選択方式のセレクトからランダム(重み付き)を選択し、どの窓口へお客を並ばせるか、確率的に制御することができます。ただし、どちらの場合も窓口に係員がおらず、閉まっている場合はお客はその窓口には並びません。

この部品内特有の時間経過はなく、入力ポートからお客アイテムを受信した順に窓口が発生している待ち行列の長さを評価し、その評価結果に応じて受信したお客アイテムを即座に出力ポートに送ります。

18.3.4 窓口選択部品:切符購入窓口 (SelectCounterServiceSingleLine)



図 18.6: 窓口選択部品 : 切符購入窓口

部品「窓口選択部品:切符購入窓口」(図 18.6) は、入力されたお客アイテムを出力ポートに送信します。この出力ポートはいわば行列であり、窓口が利用可能になり次第、このポートに留まっているお客アイテムは窓口に送られます。

18.3.5 窓口利用部品：レジ (UseCounterServiceMultipleLines)



図 18.7: 窓口利用部品：レジ

部品「窓口利用部品：レジ」(図 18.7) は、窓口 (18.1.1 章) を指定した時間だけ利用します。

窓口

利用する窓口を表します。

サービス時間

取得した窓口を利用する時間を表します。

18.3.6 窓口利用部品:切符購入窓口 (UseCounterServiceSingleLine)



図 18.8: 窓口利用部品：切符購入窓口

部品「窓口利用部品：切符購入窓口」(図 18.7) は、窓口 (18.1.1 章) を指定した時間だけ利用します。

窓口

利用する窓口を表します。

サービス時間

取得した窓口を利用する時間を表します。

第19章 乱数

乱数属性について説明します。

19.1 乱数属性

部品編集画面の属性設定タブ (6.1.2 章) や、フローアイテム編集画面の属性設定タブ (9.1.2 章) では乱数属性を設定出来ます (図 19.1)。乱数を設定する事で、フローアイテム生成時の生成間隔やフローアイテム自身のある属性を、理論分布に従った乱数にする事が出来ます。または、実際に発生したデータを元にサンプリングを行う、あるいは、そのまま再生する事で、より正確なモデル化も行えます。



図 19.1: 乱数属性

なお、時計のマークをクリックすると、時間単位を選べます (図 19.2)。もし、指定したい値が時間の場合、指定した値の単位を指定する事ができます。もし、値が時間を表している場合で、時間単位を指定しなかった場合、その値の単位は暗黙的に秒であると解釈されます。値の単位が時間でない場合、時間単位を指定しないで下さい。



図 19.2: 時間単位を指定した乱数属性

一番左の生成方式を変更するとその乱数に応じた引数が表示されます。乱数属性には以下があります。

19.1.1 固定

常に同じ値を返します。

型

実数、整数、文字列、無変換、空リストを選ぶ事が出来ます。

値

返す値を指定します。

19.1.2 指数分布

指数分布に従う乱数を返します。

平均

指数分布の平均パラメータを指定します。

乱数の種

グローバル系列を指定すると、グローバル系列から乱数を取
得します。独自系列を指定すると指定した値によって定められる
一意な独自系列から乱数を取得します。

確率密度関数

平均を μ とすると、確率密度関数は以下になります。

$$f(x; \mu) = \frac{e^{-\frac{x}{\mu}}}{\mu} \quad (x \geq 0, \mu > 0)$$

19.1.3 正規分布

正規分布に従う乱数を返します。

平均

指数分布の平均パラメータを指定します。

標準偏差

指数分布の標準偏差パラメータを指定します。

乱数の種

グローバル系列を指定すると、グローバル系列から乱数を取
得します。独自系列を指定すると指定した値によって定められる
一意な独自系列から乱数を取得します。

確率密度関数

平均を μ 、標準偏差を σ とすると、確率密度関数は以下になります。

$$f(x; \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (\sigma > 0)$$

19.1.4 対数正規分布

対数正規分布に従う乱数を返します。

平均

指数分布の平均パラメータを指定します。

標準偏差

指数分布の標準偏差パラメータを指定します。

乱数の種

グローバル系列を指定すると、グローバル系列から乱数を取
得します。独自系列を指定すると指定した値によって定められる
一意な独自系列から乱数を取得します。

確率密度関数

平均を μ 、標準偏差を σ とすると、確率密度関数は以下になります。

$$f(x; \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma x} e^{-\frac{(\log x - \mu)^2}{2\sigma^2}} \quad (\sigma > 0)$$

19.1.5 一様分布

一様分布に従う乱数を返します。

最小

一様分布の最小パラメータを指定します。

最大

一様分布の最大パラメータを指定します。

乱数の種

グローバル系列を指定すると、グローバル系列から乱数を取
得します。独自系列を指定すると指定した種による独自系列から
乱数を取得します。

確率密度関数

最小を a 、最大を b とすると、確率密度関数は以下になります。

$$f(x; a, b) = \frac{1}{b - a} \quad (a \leq x < b)$$

19.1.6 ベータ分布

ベータ分布に従う乱数を返します。

形状 1

ベータ分布の形状 1 パラメータを指定します。

形状 2

ベータ分布の形状 2 パラメータを指定します。

乱数の種

グローバル系列を指定すると、グローバル系列から乱数を取
得します。独自系列を指定すると指定した値によって定められる
一意な独自系列から乱数を取得します。

確率密度関数

形状 1 を α 、形状 2 を β とすると、確率密度関数は以下になります。

$$f(x; \alpha, \beta) = \frac{1}{B(\alpha, \beta)} x^{\alpha-1} (1-x)^{\beta-1} \quad (0 \leq x \leq 1, \alpha > 0, \beta > 0)$$

19.1.7 ガンマ分布

ガンマ分布に従う乱数を返します。

形状

ガンマ分布の形状パラメータを指定します。

尺度

ガンマ分布の尺度パラメータを指定します。

乱数の種

グローバル系列を指定すると、グローバル系列から乱数を取
得します。独自系列を指定すると指定した値によって定められる
一意な独自系列から乱数を取得します。

確率密度関数

形状を α 、尺度を β とすると、確率密度関数は以下になります。

$$f(x; \alpha, \beta) = x^{\alpha-1} \frac{e^{-x/\beta}}{\Gamma(\alpha) \beta^\alpha} \quad (x \geq 0, \alpha > 0, \beta > 0)$$

19.1.8 アーラン分布

アーラン分布に従う乱数を返します。

正整数

アーラン分布の形状パラメータを指定します。

β

アーラン分布の尺度パラメータを指定します。

確率密度関数

形状を α 、尺度を β とすると、確率密度関数は以下になります。

$$f(x; \alpha, \beta) = x^{\alpha-1} \frac{e^{-x/\beta}}{(\alpha-1)! \beta^\alpha} \quad (x \geq 0, \alpha > 0, \alpha \in \mathbf{N}, \beta > 0)$$

19.1.9 パレート分布

パレート分布に従う乱数を返します。

形状

パレート分布の形状パラメータを指定します。

最小値

パレート分布の最小値パラメータを指定します。

乱数の種

グローバル系列を指定すると、グローバル系列から乱数を取
得します。独自系列を指定すると指定した値によって定められる
一意な独自系列から乱数を取得します。

確率密度関数

形状を a 、最小値を b とすると、確率密度関数は以下になります。

$$f(x; a, b) = \frac{ab^a}{x^{a+1}} \quad (x \geq b, a > 0, b > 0)$$

19.1.10 ワイブル分布

ワイブル分布に従う乱数を返します。

形状

ワイブル分布の形状パラメータを指定します。

尺度

ワイブル分布の尺度パラメータを指定します。

乱数の種

グローバル系列を指定すると、グローバル系列から乱数を取
得します。独自系列を指定すると指定した値によって定められる
一意な独自系列から乱数を取得します。

確率密度関数

形状を α 、尺度を β とすると、確率密度関数は以下になります。

$$f(x; \alpha, \beta) = \frac{\alpha}{\beta} \left(\frac{x}{\beta}\right)^{\alpha-1} e^{-\left(\frac{x}{\beta}\right)^\alpha} \quad (x \geq 0, \alpha > 0, \beta > 0)$$

19.1.11 カイ二乗分布

カイ二乗分布に従う乱数を返します。

自由度

カイ二乗分布の自由度パラメータを指定します。

乱数の種

グローバル系列を指定すると、グローバル系列から乱数を取
得します。独自系列を指定すると指定した値によって定められる
一意な独自系列から乱数を取得します。

確率密度関数

自由度を m とすると、確率密度関数は以下になります。

$$f(x; m) = \frac{(1/2)^{m/2}}{\Gamma(m/2)} x^{m/2-1} e^{-x/2} \quad (m > 0)$$

19.1.12 F 分布

F 分布に従う乱数を返します。

自由度 1

F 分布の分子の自由度パラメータを指定します。

自由度 2

F 乗分布の分母の自由度パラメータを指定します。

乱数の種

グローバル系列を指定すると、グローバル系列から乱数を取
得します。独自系列を指定すると指定した値によって定められる
一意な独自系列から乱数を取得します。

確率密度関数

自由度 1 を m_1 、自由度 2 を m_2 とすると、確率密度関数は以下になります。

$$f(x; m_1, m_2) = \frac{\Gamma(\frac{m_1+m_2}{2}) (\frac{m_1}{m_2})^{\frac{m_1}{2}} x^{\frac{m_1}{2}-1}}{\Gamma(\frac{m_1}{2}) \Gamma(\frac{m_2}{2}) (1 + \frac{m_1}{m_2} x)^{\frac{m_1+m_2}{2}}} \quad (m_1 > 0, m_2 > 0)$$

19.1.13 ロジスティック分布

ロジスティック分布に従う乱数を返します。

位置

ロジスティック分布の位置パラメータを指定します。

尺度

ロジスティック分布の尺度パラメータを指定します。

乱数の種

グローバル系列を指定すると、グローバル系列から乱数を取
得します。独自系列を指定すると指定した値によって定められる
一意な独自系列から乱数を取得します。

確率密度関数

位置を μ 、尺度を s とすると、確率密度関数は以下になります。

$$f(x; \mu, s) = \frac{e^{-(x-\mu)/s}}{s(1 + e^{-(x-\mu)/s})^2}$$

19.1.14 非心カイ二乗分布

非心カイ二乗分布に従う乱数を返します。

自由度

非心カイ二乗分布の自由度パラメータを指定します。

非心度

非心カイ二乗分布の非心度パラメータを指定します。

乱数の種

グローバル系列を指定すると、グローバル系列から乱数を取
得します。独自系列を指定すると指定した値によって定められる
一意な独自系列から乱数を取得します。

確率密度関数

自由度を m 、非心度を δ とすると、確率密度関数は以下になります。

$$f(x; m, \delta) = \frac{e^{-\frac{x+\delta}{2}}}{2^{\frac{m}{2}}} \sum_{j=0}^{\infty} \left(\frac{\delta}{4}\right)^j \frac{x^{\frac{m}{2}+j-1}}{j! \Gamma(\frac{m}{2} + j)} \quad (m > 0, \delta \geq 0)$$

19.1.15 非心 F 分布

非心 F 分布に従う乱数を返します。

自由度 1

非心 F 分布の分子の自由度パラメータを指定します。

自由度 2

非心 F 分布の分母の自由度パラメータを指定します。

非心度

非心 F 分布の非心度パラメータを指定します。

乱数の種

グローバル系列を指定すると、グローバル系列から乱数を取
得します。独自系列を指定すると指定した値によって定められる
一意な独自系列から乱数を取得します。

確率密度関数

自由度 1 を m_1 、自由度 2 を m_2 、非心度を δ とすると、確率密度関数は以
下になります。

$$f(x; m_1, m_2, \delta) = \sum_{j=0}^{\infty} \frac{e^{-\frac{\delta}{2}} \left(\frac{\delta}{2}\right)^j}{j!} \frac{(m_1 + 2j)^{\frac{m_1+2j}{2}} m_2^{\frac{m_2}{2}}}{B\left(\frac{m_1+2j}{2}, \frac{m_2}{2}\right)} x^{\frac{m_1+2j}{2}-1} (m_2 + (m_1+2j)x)^{-\frac{(m_1+2j)+m_2}{2}}$$

19.1.16 コーシー分布

コーシー分布に従う乱数を返します。

乱数の種

グローバル系列を指定すると、グローバル系列から乱数を取
得します。独自系列を指定すると指定した値によって定められる
一意な独自系列から乱数を取得します。

確率密度関数

$$f(x) = \frac{1}{\pi(1+x^2)}$$

19.1.17 レイリー分布

レイリー分布に従う乱数を返します。

尺度

レイリー分布の尺度パラメータを指定します。

乱数の種

グローバル系列を指定すると、グローバル系列から乱数を取
得します。独自系列を指定すると指定した値によって定められる
一意な独自系列から乱数を取得します。

確率密度関数

尺度を θ とすると、確率密度関数は以下になります。

$$f(x; \theta) = \frac{x}{\theta^2} e^{-\frac{x^2}{2\theta^2}}$$

19.1.18 t 分布

t 分布に従う乱数を返します。

自由度

t 分布の自由度パラメータを指定します。

乱数の種

グローバル系列を指定すると、グローバル系列から乱数を取
得します。独自系列を指定すると指定した値によって定められる
一意な独自系列から乱数を取得します。

確率密度関数

自由度を m とすると、確率密度関数は以下になります。

$$f(x; m) = \frac{\Gamma((m+1)/2)}{\sqrt{m\pi} \Gamma(m/2)} (1 + x^2/m)^{-(m+1)/2}$$

19.1.19 三角分布

三角分布に従う乱数を返します。

最小値

三角分布の最小値パラメータを指定します。

最頻値

三角分布の最頻値パラメータを指定します。

最大値

三角分布の最大値パラメータを指定します。

乱数の種

グローバル系列を指定すると、グローバル系列から乱数を取
得します。独自系列を指定すると指定した値によって定められる
一意な独自系列から乱数を取得します。

確率密度関数

最小値を α 、最頻値を γ 、最大値を β とすると、確率密度関数は以下になります。

$$f(x; \alpha, \gamma, \beta) = \begin{cases} \frac{2(x-\alpha)}{(\beta-\alpha)(\gamma-\alpha)} & (\alpha \leq x \leq \gamma) \\ \frac{2(\beta-x)}{(\beta-\alpha)(\beta-\gamma)} & (\gamma \leq x \leq \beta) \end{cases}$$

19.1.20 二項分布

二項分布に従う乱数を返します。

n

二項分布の試行回数パラメータを指定します。

確率

二項分布の確率パラメータを指定します。

乱数の種

グローバル系列を指定すると、グローバル系列から乱数を取
得します。独自系列を指定すると指定した値によって定められる
一意な独自系列から乱数を取得します。

確率関数

試行回数を n 、確率を p とすると、確率関数は以下になります。

$$f(k; n, p) = \binom{n}{k} p^k (1-p)^{n-k} \quad (n > 0, 0 \leq p \leq 1)$$

19.1.21 幾何分布

幾何分布に従う乱数を返します。

確率

幾何分布の確率パラメータを指定します。

乱数の種

グローバル系列を指定すると、グローバル系列から乱数を取
得します。独自系列を指定すると指定した値によって定められる
一意な独自系列から乱数を取得します。

確率関数

確率を p とすると、確率関数は以下になります。

$$f(k; p) = (1 - p)^{k-1} p \quad (0 \leq k < \infty)$$

19.1.22 超幾何分布

超幾何分布に従う乱数を返します。

m

超幾何分布のある属性を持つ要素数パラメータを指定します。

n

超幾何分布のそれ以外の要素数パラメータを指定します。

k

超幾何分布のサンプル数パラメータを指定します。

乱数の種

グローバル系列を指定すると、グローバル系列から乱数を取
得します。独自系列を指定すると指定した値によって定められる
一意な独自系列から乱数を取得します。

確率関数

ある属性を持つ要素数 m 、それ以外の要素数を n 、サンプル数を N とす
ると、確率関数は以下になります。

$$f(x; n, m, N) = \frac{\binom{m}{x} \binom{n}{N-x}}{\binom{m+n}{N}} \quad (0 \leq x \leq m, n+m-N \leq x \leq n)$$

19.1.23 負の二項分布

負の二項分布に従う乱数を返します。

n

負の二項分布の試行回数パラメータを指定します。

確率

負の二項分布の確率パラメータを指定します。

乱数の種

グローバル系列を指定すると、グローバル系列から乱数を取
得します。独自系列を指定すると指定した値によって定められる
一意な独自系列から乱数を取得します。

確率関数

試行回数を n 、確率を p とすると、確率関数は以下になります。

$$f(x; n, p) = \binom{n+x-1}{n-1} p^n (1-p)^x \quad (n > 0, 0 \leq p \leq 1)$$

19.1.24 ポアソン分布

ポアソン分布に従う乱数を返します。

平均

ポアソン分布の平均パラメータを指定します。

乱数の種

グローバル系列を指定すると、グローバル系列から乱数を取
得します。独自系列を指定すると指定した値によって定められる
一意な独自系列から乱数を取得します。

確率関数

平均を λ とすると、確率関数は以下になります。

$$f(x; \lambda) = \frac{\lambda^x e^{-\lambda}}{x!}$$

19.1.25 経験分布

経験分布に従う乱数を返します。

型

実数、整数、文字列、無変換を選ぶ事が出来ます。

重み表

重みと値のリストを指定します。

乱数の種

グローバル系列を指定すると、グローバル系列から乱数を取
得します。独自系列を指定すると指定した値によって定められる
一意な独自系列から乱数を取得します。

重み表の行ラベルを右クリックするとメニューが表示されます。(図 19.3)

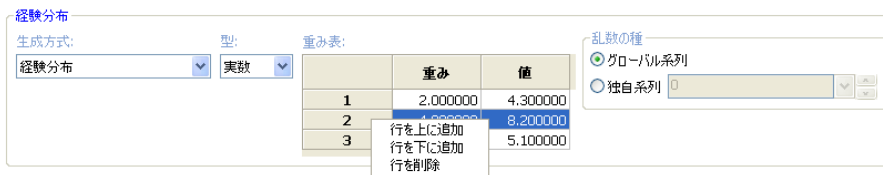


図 19.3: 経験分布重み表

行を上へ追加、行を削除、行を下へ追加する事が出来ます。

確率関数

重みを $\{w_i\}$ 、値を $\{v_i\}$ とすると、確率関数は以下になります。

$$f(v_i; \{w_j\}) = \frac{w_i}{\sum_j w_j} \quad (w_i \geq 0)$$

19.1.26 経験分布 (列指定、重み無し)

入力フォルダにあるデータを利用した重み無し経験分布に従う乱数 (サン
プリング) を返します。

値列

入力フォルダにあるモニターの列を選ぶ事が出来ます。時系列モニターの列は表示されません。

乱数の種

グローバル系列を指定すると、グローバル系列から乱数を取ります。独自系列を指定すると指定した値によって定められる一意な独自系列から乱数を取得します。

値列は図 19.4 のようにメニューから選択する事が出来ます。外部データのインポートについては、4.4 章を参照して下さい。

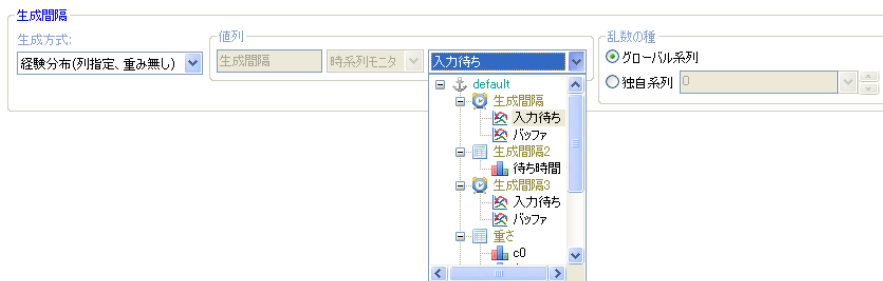


図 19.4: 列選択メニュー

確率関数

値列を $\{v_i\}$ とすると、確率関数は以下になります。

$$f(v_i) = \frac{1}{\|\{v_i\}\|}$$

19.1.27 経験分布 (列指定、重み有り)

入力フォルダにあるデータを利用した重み付き経験分布に従う乱数を返します。

重み列

入力フォルダにあるモニターの列を選ぶ事が出来ます。時系列モニターの列は表示されません。

値列

入力フォルダにあるモニターの列を選ぶ事が出来ます。時系列モニターの列は表示されません。

乱数の種

グローバル系列を指定すると、グローバル系列から乱数を取ります。独自系列を指定すると指定した値によって定められる一意な独自系列から乱数を取得します。

重み列、値列は図 19.5 のようにメニューから選択する事が出来ます。外部データのインポートについては、4.4 章を参照して下さい。

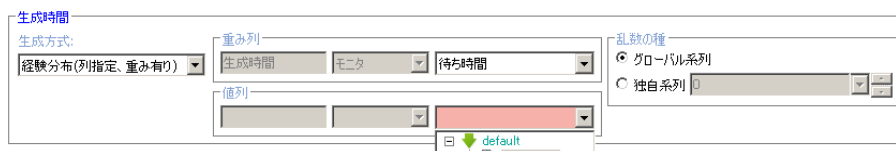


図 19.5: 列選択メニュー

確率関数

重み列を $\{w_i\}$ 、値列を $\{v_i\}$ とすると、確率関数は以下になります。

$$f(v_i; \{w_j\}) = \frac{w_i}{\sum_j w_j} \quad (w_i \geq 0)$$

19.1.28 経験分布 (列指定、時系列)

入力フォルダにあるデータを利用した重み付き経験分布に従う乱数を返します。

値列

入力フォルダにある時系列モニターの列を選ぶ事が出来ます。モニターの列は指定出来ません。

乱数の種

グローバル系列を指定すると、グローバル系列から乱数を取ります。独自系列を指定すると指定した値によって定められる一意な独自系列から乱数を取得します。

値列は図 19.6 のようにメニューから選択する事が出来ます。外部データのインポートについては、4.4 章を参照して下さい。

19.1 乱数属性

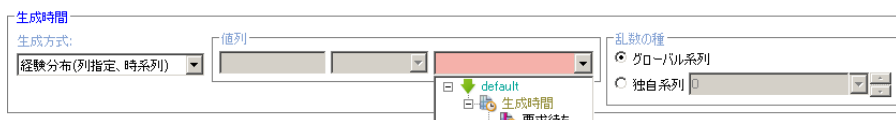


図 19.6: 列選択メニュー

確率関数

値列を $(\{v_i\}, \{t_i\})$ とすると、確率関数は以下になります。

$$w_j = \Delta t_j = t_{j+1} - t_j \quad (0 \leq j < \|\{v_i\}\|)$$
$$f(v_i; \{w_j\}) = \frac{w_i}{\sum_j w_j} \quad (w_i \geq 0, 0 \leq i < \|\{v_i\}\|)$$

19.1.29 密度分布 (列指定、重み無し)

入力フォルダにあるデータを利用した密度分布に従う乱数を返します。

値列

入力フォルダにあるモニターの列を選ぶ事が出来ます。時系列モニターの列は表示されません。

乱数の種

グローバル系列を指定すると、グローバル系列から乱数を取ります。独自系列を指定すると指定した値によって定められる一意な独自系列から乱数を取得します。

値列は図 19.7 のようにメニューから選択する事が出来ます。外部データのインポートについては、4.4 章を参照して下さい。

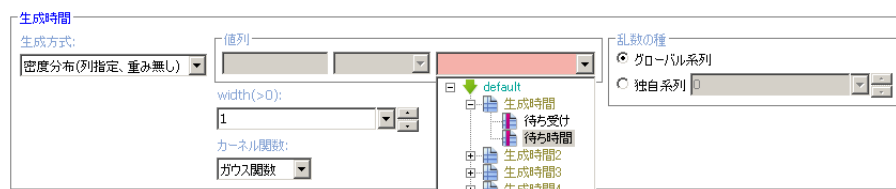


図 19.7: 列選択メニュー

確率関数

値列を $\{v_i\}$ とすると、確率関数は以下になります。カーネル関数を K とすると、カーネル密度推定では以下の式により密度を推定します。

$$f(x) = \sum_{i=1}^N K\left(\frac{x - v_i}{\omega}\right)$$

ここで ω は間隔幅 (width) です。

カーネル関数は、それぞれ以下で定義されます。

- ガウス関数

$$f(x; \mu, \omega) = \frac{1}{\sqrt{2\pi\omega}} e^{-\frac{(x-\mu)^2}{2\omega^2}} \quad (\omega > 0)$$

- 三角関数

$$f(x; \mu, \omega) = \frac{1}{\omega^2}(x - \mu) + \frac{1}{\omega} \quad (\omega > 0, \mu - \omega \leq x \leq \mu + \omega)$$

- 矩形関数

$$f(x; \mu, \omega) = \frac{1}{2\omega} \quad (\omega > 0, \mu - \omega \leq x \leq \mu + \omega)$$

- コサイン関数

$$f(x; \mu, \omega) = \frac{1}{2\omega} \left[1 + \cos\left(\frac{\pi(x - \mu)}{\omega}\right) \right] \quad (\omega > 0, \mu - \omega \leq x \leq \mu + \omega)$$

19.1.30 密度分布 (列指定、重み有り)

入力フォルダにあるデータを利用した重み付き密度分布に従う乱数を返します。

重み列

入力フォルダにあるモニターの列を選ぶ事が出来ます。時系列モニターの列は表示されません。

値列

入力フォルダにあるモニターの列を選ぶ事が出来ます。時系列モニターの列は表示されません。

乱数の種

グローバル系列を指定すると、グローバル系列から乱数を取ります。独自系列を指定すると指定した値によって定められる一意な独自系列から乱数を取ります。

重み列、値列は図 19.8 のようにメニューから選択する事が出来ます。外部データのインポートについては、4.4 章を参照して下さい。

19.1 乱数属性

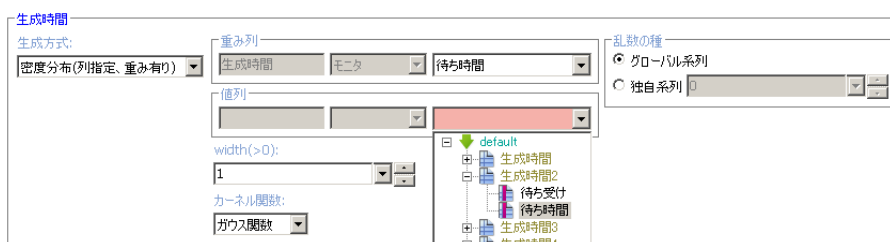


図 19.8: 列選択メニュー

確率関数

重み列を $\{w_i\}$ 、値列を $\{v_i\}$ とすると、確率関数は以下になります。カーネル関数を K とすると、カーネル密度推定では以下の式により密度を推定します。

$$f(x) = \frac{\sum_{i=1}^N w_i K\left(\frac{x - v_i}{\omega}\right)}{\sum_j w_j}$$

ここで ω は間隔幅 (width) です。

カーネル関数は、それぞれ以下で定義されます。

- ガウス関数

$$f(x; \mu, \omega) = \frac{1}{\sqrt{2\pi\omega}} e^{-\frac{(x-\mu)^2}{2\omega^2}} \quad (\omega > 0)$$

- 三角関数

$$f(x; \mu, \omega) = \frac{1}{\omega^2}(x - \mu) + \frac{1}{\omega} \quad (\omega > 0, \mu - \omega \leq x \leq \mu + \omega)$$

- 矩形関数

$$f(x; \mu, \omega) = \frac{1}{2\omega} \quad (\omega > 0, \mu - \omega \leq x \leq \mu + \omega)$$

- コサイン関数

$$f(x; \mu, \omega) = \frac{1}{2\omega} \left[1 + \cos\left(\frac{\pi(x - \mu)}{\omega}\right) \right] \quad (\omega > 0, \mu - \omega \leq x \leq \mu + \omega)$$

19.1.31 密度分布 (列指定、時系列)

入力フォルダにあるデータを利用した密度分布に従う乱数を返します。

値列

入力フォルダにあるモニターの列を選ぶ事が出来ます。時系列モニターの列は表示されません。

乱数の種

グローバル系列を指定すると、グローバル系列から乱数を取
得します。独自系列を指定すると指定した値によって定められる
一意な独自系列から乱数を取得します。

値列は図 19.9 のようにメニューから選択する事が出来ます。外部データの
インポートについては、4.4 章を参照して下さい。

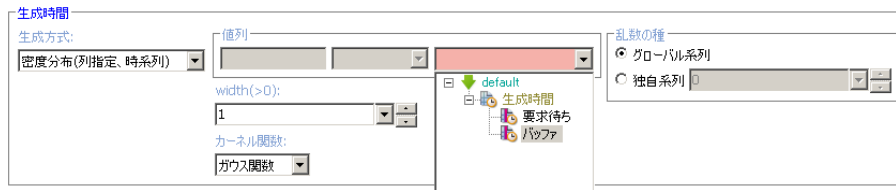


図 19.9: 列選択メニュー

確率関数

値列を $(\{v_i\}, \{t_i\})$ とすると、確率関数は以下になります。カーネル関数を
 K とすると、カーネル密度推定では以下の式により密度を推定します。

$$w_j = \Delta t_j = t_{j+1} - t_j \quad (0 \leq j < \|\{v_i\}\|)$$

$$f(x) = \frac{\sum_{i=1}^N w_i}{\sum_j w_j} K\left(\frac{x - v_i}{\omega}\right)$$

ここで ω は間隔幅 (width) です。

カーネル関数は、それぞれ以下で定義されます。

- ガウス関数

$$f(x; \mu, \omega) = \frac{1}{\sqrt{2\pi\omega}} e^{-\frac{(x-\mu)^2}{2\omega^2}} \quad (\omega > 0)$$

- 三角関数

$$f(x; \mu, \omega) = \frac{1}{\omega^2}(x - \mu) + \frac{1}{\omega} \quad (\omega > 0, \mu - \omega \leq x \leq \mu + \omega)$$

- 矩形関数

$$f(x; \mu, \omega) = \frac{1}{2\omega} \quad (\omega > 0, \mu - \omega \leq x \leq \mu + \omega)$$

- コサイン関数

$$f(x; \mu, \omega) = \frac{1}{2\omega} \left[1 + \cos\left(\frac{\pi(x - \mu)}{\omega}\right) \right] \quad (\omega > 0, \mu - \omega \leq x \leq \mu + \omega)$$

19.1.32 再生 (列指定)

入力フォルダにあるデータをそのまま再生します。

値列

入力フォルダにあるモニターあるいは時系列モニターの列を選ぶ事が出来ます。

繰り返す

再生を繰り返すかどうかを指定します。

部品等の乱数属性として使う場合は繰り返さないと実行時にエラーが発生するので注意が必要です。

値列は図 19.10 のようにメニューから選択する事が出来ます。外部データのインポートについては、4.4 章を参照下さい。

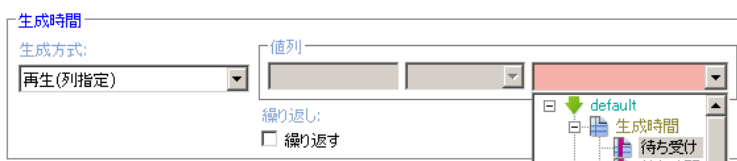


図 19.10: 列選択メニュー

19.1.33 再生 (ステップ)

開始点から終了点までをステップ置きに再生します。

型

実数、整数を選ぶ事が出来ます。

開始

開始点を指定します。

終了

終了点を指定します。この値は含まれません。終了点は開始点以上になるよう設定して下さい。

ステップ

ステップを指定します。正の値になるように設定してください。

繰り返す

繰り返して再生するかを指定します。

部品等の乱数属性として使う場合は繰り返さないと実行時にエラーが発生するので注意が必要です。

19.1.34 再生 (変化率ステップ)

基準値からの変化率により動かし方を指定します。具体的には、基準値を b 、変化率 (上限) を α (パーセント)、変化率 (下限) を γ (パーセント)、ステップを δ (パーセント) とすると、次の値を動きます： $b(1 + \frac{\gamma}{100})$, $b(1 + \frac{\delta+\gamma}{100})$, $b(1 + \frac{2\delta+\gamma}{100})$, ..., $b(1 + \frac{\alpha}{100})$.

型

実数、整数を選ぶ事が出来ます。

基準値

基準値を設定します。

変化率 (上限)

変化率の上限を指定します。

変化率 (下限)

変化率の下限を指定します。

ステップ

ステップを指定します。正の値になるように設定してください。

繰り返す

繰り返して再生するかを指定します。

部品等の乱数属性として使う場合は繰り返さないと実行時にエラーが発生するので注意が必要です。

19.1.35 ユーザ定義ジェネレータ

ユーザ定義のジェネレータを指定する事が出来ます。

ジェネレータ

ユーザ定義のジェネレータを指定します。

繰り返し無しの再生の例

```
コード例  
iter([1, 3, 7, 8])
```

と書けば、1, 3, 7, 8 をこの順に生成します。

繰り返し有りの再生の例

カスタムコード編集 (5.5 章) で、

```
コード例  
def usergen():  
    while True:  
        yield 1  
        yield 3  
        yield 7  
        yield 8
```

とジェネレータ `usergen` を定義し、ユーザ定義ジェネレータを選択しジェネレータに `usergen()` と書けば、1, 3, 7, 8 を繰り返し生成します。

あるいは、カスタムコード編集で、

```
コード例  
import itertools
```

と書き、ジェネレータとして、`itertools.cycle([1, 3, 7, 9])` と書いても同じ動作をします。

乱数の組み合わせの例

以下のようなジェネレータを定義すれば、より複雑な乱数を生成する事が出来ます。

コード例

```
def myrandom():
    g1 = uniformDistribution(0, 1)
    g2 = normalDistribution(100, 3)
    g3 = exponentialDistribution(5)
    while True:
        yield next(g1) * next(g2) + next(g3)
```

psim 言語での乱数の生成については psim 言語リファレンスマニュアルを参照して下さい。

時間によって分布を変化させる例

以下のようなジェネレータを定義すれば、時間によって分布を変化させる事が出来ます。

コード例

```
def myrandom():
    g1 = uniformDistribution(0, 1)
    g2 = normalDistribution(100, 3)
    g3 = exponentialDistribution(5)
    while True:
        if now() < 500:
            yield next(g1)
        elif now () < 1000:
            yield next(g2)
        else:
            yield next(g3)
```

第20章 キュー

資源やポートで設定するキューについてを説明します。

キューには FIFO キュー, 優先度付きキュー、スタック、ランダムがあります。資源編集画面によって指定出来るキューの個数はそれぞれ違い、それぞれのキューの役割りは異なります (図 20.1)。またポートのキューは暗黙的に FIFO キューになります。

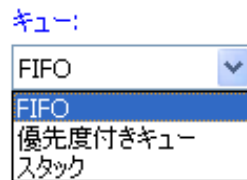


図 20.1: キュー選択メニュー

20.1 キューの種別

20.1.1 FIFO キュー

先入れ、先出しのキューで。First In, First Out の略です。1, 2, 3 の順番で入力された場合、1, 2, 3 で出力します (図 20.2)。例えば銀行の窓口の行列など。

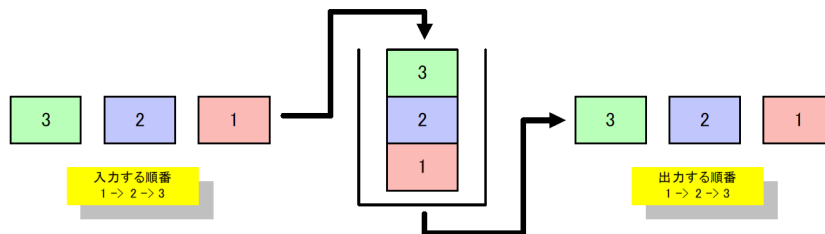


図 20.2: FIFO キュー

20.1.2 優先度付きキュー

優先度付きキューです。入力された順番とは関係なく、優先度 priority の高い順番に出力します (図 20.3)。例えば、ある作業者がいて、仕事を割り振られますが、仕事には優先度があり優先度の高い順に作業する場合など。

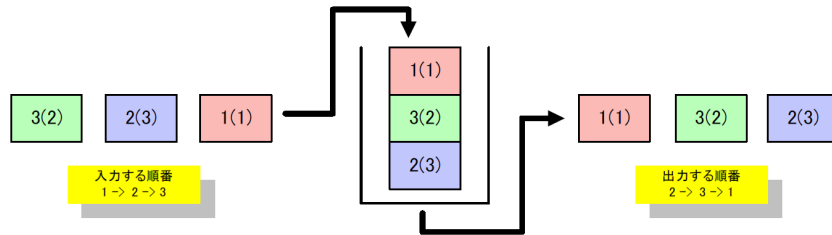


図 20.3: 優先度付きキュー (括弧は優先度)

20.1.3 スタック

後入れ、先出しのキューで。LIFO (Last In, First Out) キューとも言います。1, 2, 3 の順番で入力された場合、3, 2, 1 で出力します。(図 20.4) 例えば、ある生産ラインにおいて、上流工程から流れてきた部品が重ねられていき、一番上に重ねられた部品から次の工程に流れて行く場合など。

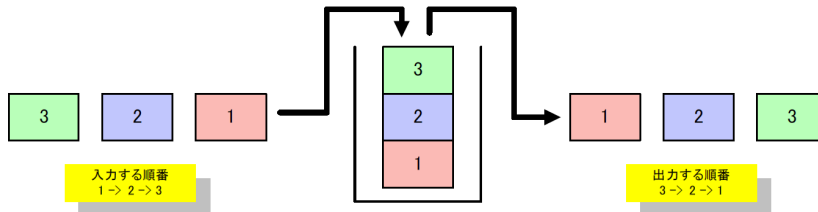


図 20.4: スタック

第21章 グラフ

S⁴ Simulation System では多彩なグラフを容易に素早く作成し、シミュレーション結果を多面的に把握し分析する機能を提供します。

グラフ機能として、定型グラフ、グラフ部品、リアルタイムグラフ部品の3つの機能があります。定型グラフ機能は、定型的なグラフを作成する機能です。グラフ部品は、さまざまなグラフで可視化する機能です。リアルタイムグラフ部品は、シミュレーション実行中の結果を可視化する機能です。

21.1 定型グラフ

定型グラフ機能には、ワークスペースタブの出力フォルダ以下にあるモニターまたは時系列モニターのコンテキストメニュー (図 21.1) からアクセスします。

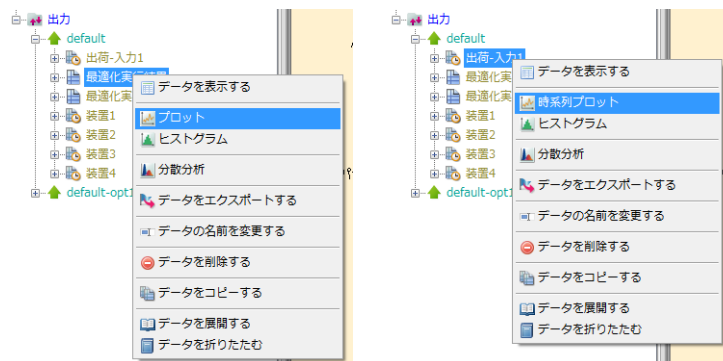


図 21.1: モニター・時系列モニターのコンテキストメニュー

モニターは「プロット」(図 21.2)、「ヒストグラム」(図 21.3)をメニューから作成できます。「プロット」はモニターの数値列を対象にして折れ線グラフを作成します。縦軸が値、横軸が順番に対応します。「ヒストグラム」は、モニターの数値列を対象にしてヒストグラムを作成します。

第 21 章 グラフ

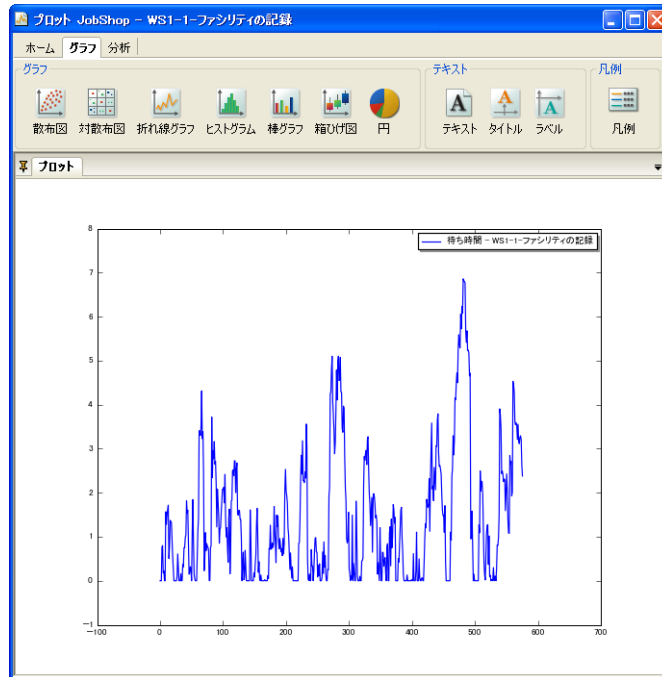


図 21.2: モニターのプロット

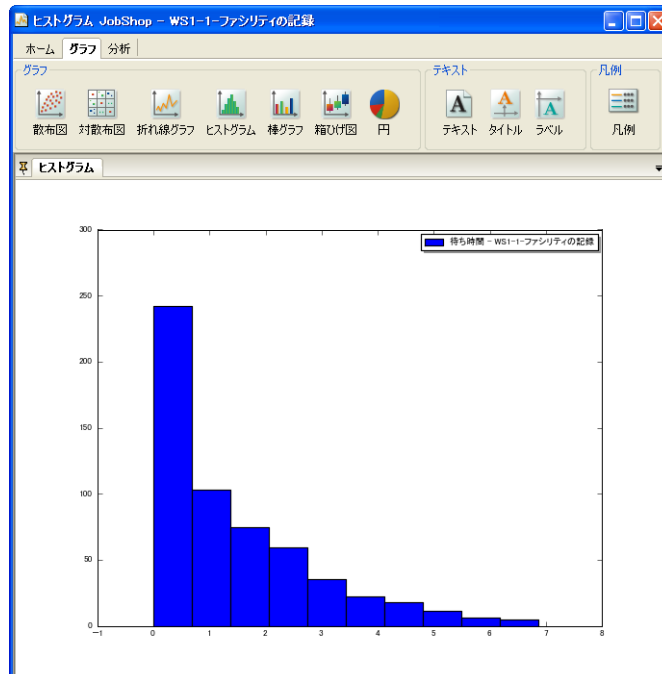


図 21.3: モニターのヒストグラム

21.1 定型グラフ

時系列モニターは「時系列プロット」(図 21.4)、「ヒストグラム」(図 21.5)をメニューから作成できます。「時系列プロット」は時系列モニターの数値列を対象にして折れ線グラフを作成します。縦軸が値、横軸が時間に対応します。「ヒストグラム」は時系列モニターの数値列を対象にして時間差を重みとしたヒストグラムを作成します。

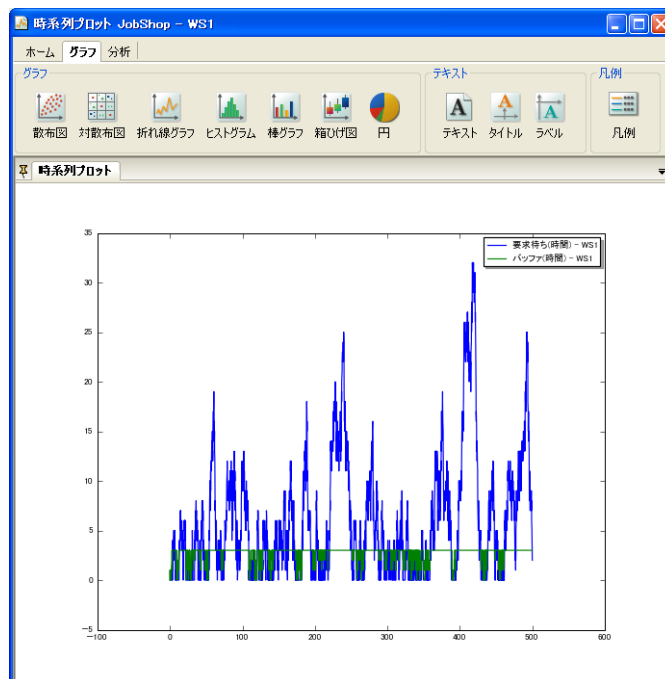


図 21.4: 時系列モニターの時系列プロット

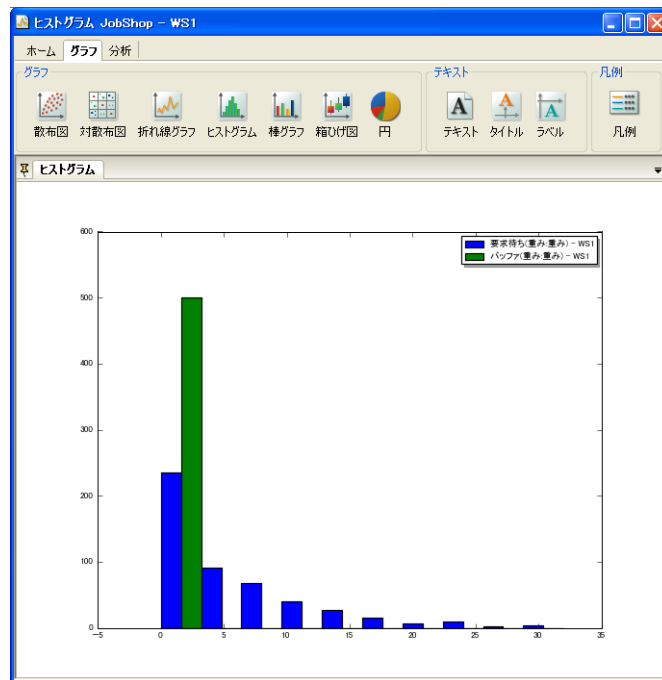


図 21.5: 時系列モニターのヒストグラム

定型グラフは、他のグラフのように編集可能ですがグラフ部品 (21.2 章) やリアルタイムグラフ部品 (21.3 章) のようにこの設定を保存することは出来ません。

21.2 グラフ部品

グラフ部品は、モデル上に配置して使用します。グラフ部品をダブルクリックし、グラフウィンドウを開きます。グラフは、このウィンドウで作成します。

パラメータ編集 (5.4 章) で指定される出力フォルダ内にあるデータに対してグラフを作成します。指定された出力フォルダが存在しない場合、エラーメッセージ (図 21.6) が表示され、グラフ部品を編集できません。

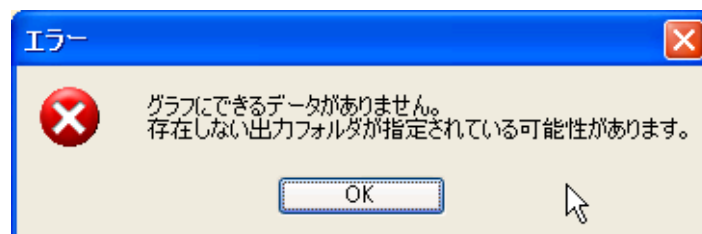


図 21.6: 出力フォルダがない場合のエラーメッセージ

この場合、シミュレーションを一度実行する、またはパラメータの出力フォルダを実際に存在する別の出力フォルダに変更することで、グラフ部品を編集できるようになります。ただし別の出力フォルダを指定した場合、対応するデータが実際にはない場合は正しくグラフが描かれませんが、データを指定しなおす必要があります。

グラフの作成方法はグラフマニュアルを参照してください。

21.3 リアルタイムグラフ部品

リアルタイムグラフ部品は、モデル上に配置して使用します。リアルタイムグラフ部品をダブルクリックし、グラフウィンドウを開きます。グラフは、このウィンドウで作成します。リアルタイムグラフ部品では、シミュレーション実行中にグラフを作成できます。

パラメータ編集 (5.4 章) で指定される出力フォルダ内にあるデータに対してグラフを作成します。指定された出力フォルダが存在しない場合、グラフ部品と同様にエラーメッセージ (図 21.6) が表示されます。

シミュレーションを一度実行する、またはパラメータの出力フォルダを実際に存在する別の出力フォルダに変更することで、グラフ部品を編集できるようになります。ただし別の出力フォルダを指定した場合、対応するデータが実際にはない場合は正しくグラフが描かれませんが。

作成されたリアルタイムグラフはシミュレーション実行時に全て表示されます。リアルタイムグラフを閉じてしまった場合、開きたいリアルタイムグラフ部品をダブルクリックすることで再度開くことができます。

グラフの作成方法はグラフマニュアルを参照してください。

第22章 グラフファイル

環境部品の中には、様々な種類のランダムグラフが用意されているが、外部からインポートしたいケースもあります。

グラフの形状のみを取り込みたい場合もあると思いますが、外部で表示上の座標を自動配置し、その情報とともに取り込みたい場合もあると思います。

環境部品が作ったランダムグラフの場合、たいていの場合、座標はランダムに配置されており、2次元上の座標に適切に配置したい場合はユーザーが手動で設定する必要があります。

もし、座標を自動的に配置したい場合は、外部のツールを利用する必要がありますが、例えば、グラフの自動配置ツールとして良く用いられているのが、以下です。

- The Open Graph Viz Platform:

- <http://gephi.org/>

Gephi には、幾つかの自動配置アルゴリズムが搭載されており、また、配置過程はリアルタイムに表示され、配置中に、様々なパラメータを変更したり、特定のノードを強制的に動かしたりする事が出来ます。(図 22.1)

22.1 グラフファイルのフォーマット

22.1.1 GEXF フォーマット

- Graph Exchange XML Format

- <http://gexf.net/format/index.html>

GEXF は、ネットワーク構造を表現する言語のひとつである。Gephi プロジェクトから発生した事もあり、Gephi との相性が非常に良い。

サンプルは以下である。

22.1 グラフファイルのフォーマット

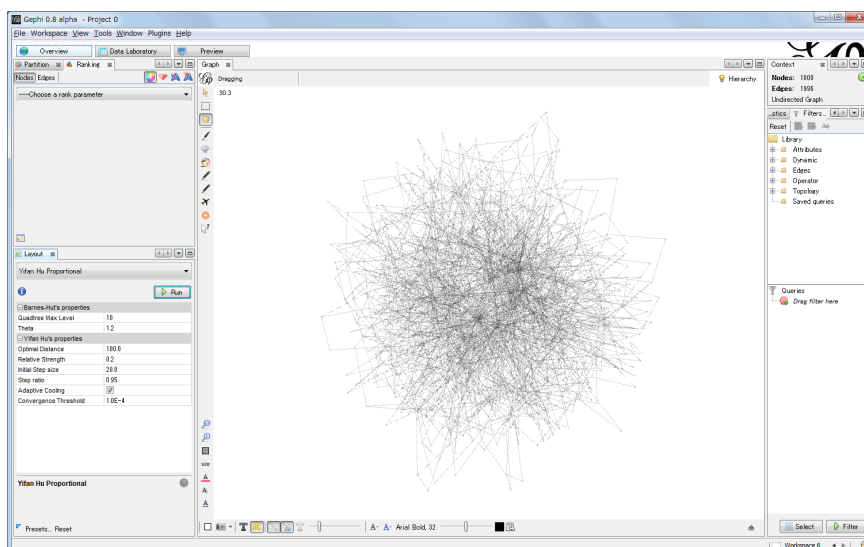


図 22.1: The Open Graph Viz Platform

コード例

```
<?xml version="1.0" encoding="UTF-8"?>
<gexf xmlns="http://www.gexf.net/1.2draft" version="1.2">
  <meta lastmodifieddate="2009-03-20">
    <creator>Gexf.net</creator>
    <description>A hello world! file</description>
  </meta>
  <graph mode="static" defaultedgetype="directed">
    <nodes>
      <node id="0" label="Hello" />
      <node id="1" label="Word" />
    </nodes>
    <edges>
      <edge id="0" source="0" target="1" />
    </edges>
  </graph>
</gexf>
```

形式は XML であり、ノードの定義部とエッジの定義部に分かれる。ノードとエッジの属性には、任意個の属性値を指定する事が出来る。

Gephi によって自動配置を行った結果のフォーマットもまた、XML の仕様として定義されている。

サンプルは以下である。

コード例

```
<?xml version="1.0" encoding="UTF-8"?>
<gexf xmlns="http://www.gexf.net/1.2draft"
      xmlns:viz="http://www.gexf.net/1.1draft/viz"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:schemaLocation="http://www.gexf.net/1.2draft
      http://www.gexf.net/1.2draft/gexf.xsd" version="1.2">
  <graph>
    <nodes>
      <node id="a" label="glossy">
        <viz:color r="239" g="173" b="66" a="0.6"/>
        <viz:position x="15.783598" y="40.109245"
                    z="0.0"/>
        <viz:size value="2.0375757"/>
        <viz:shape value="disc"/>
      </node>
    </nodes>
  </graph>
</gexf>
```

他の属性値とは独立に定義される。

22.1.2 GraphML フォーマット

- GraphML
 - <http://graphml.graphdrawing.org/>

GraphML は、ネットワーク構造を表現する言語のひとつである。

コード例

```
<?xml version="1.0" encoding="UTF-8"?>
<graphml xmlns="http://graphml.graphdrawing.org/xmlns"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://graphml.graphdrawing.org/xmlns
  http://graphml.graphdrawing.org/xmlns/1.0/graphml.xsd">
  <key id="d0" for="node" attr.name="color"
    attr.type="string">
    <default>yellow</default>
  </key>
  <key id="d1" for="edge" attr.name="weight"
    attr.type="double"/>
  <graph id="G" edgedefault="undirected">
    <node id="n0">
      <data key="d0">green</data>
    </node>
    <node id="n1"/>
    <node id="n2">
      <data key="d0">blue</data>
    </node>
    <node id="n3">
      <data key="d0">red</data>
    </node>
    <node id="n4"/>
    <node id="n5">
      <data key="d0">turquoise</data>
    </node>
    <edge id="e0" source="n0" target="n2">
      <data key="d1">1.0</data>
    </edge>
    <edge id="e1" source="n0" target="n1">
      <data key="d1">1.0</data>
    </edge>
    <edge id="e2" source="n1" target="n3">
      <data key="d1">2.0</data>
    </edge>
    <edge id="e3" source="n3" target="n2"/>
    <edge id="e4" source="n2" target="n4"/>
  </graph>
</graphml>
```

形式は XML であり、ノードの定義とエッジの定義を行う。ノードとエッジの属性には、任意個の属性値を指定する事が出来る。

Gephi によって自動配置を行った結果は、他の属性と同様に出力される。

22.2 グラフファイルの取り込み

GEXF フォーマットのファイルを取り込むには、[12.9 章](#)を参照して下さい。

GraphML フォーマットのファイルを取り込むには、[12.6 章](#)を参照して下さい。

22.3 グラフファイルの書き出し

モデル編集画面内に配置されている環境の中で、グラフ形式の部品 (末尾にグラフとついているもの) にて、右クリックして表示されるメニューから、「グラフファイルを出力する」を選んで下さい。すると、最初に、

- GEXF フォーマット
- GraphML フォーマット

を選択下さい。次ファイルダイアログが表示されるので、書き込みたいファイル名を入力して下さい。

第23章 データ表示

23.1 サマリの表示

サマリの表示方法についてを説明します。

モデルを開いている状態で、モデルメニュー (3.2.3 章) の「サマリを表示する」を選択するとサマリを表示します。あるいはツールバーのボタン (3.3.6 章)、モデル編集画面の部品がない場所で右クリックして表示されるメニュー (3.5.1 章) から実行出来ます。

サマリ表示画面 (図 23.1) では、現在選択されている出力ポートの基本統計量の一覧が表示されます。

名前	観測数	平均	標準偏差	95%信...	95%信...	変動係数	最小値	25%値	中央値	75%値	最大値	レンジ
WS4-3-ファシリティの記録	14	0.45927	0.603758	0.110671	0.807869	1.3146	0	0	0.140061	0.798347	1.69278	1.69278
待ち受け	4	---	---	---	---	---	---	---	---	---	---	---
待ち時間	4	0.423206	0.696546	-0.685154	1.53157	1.64588	0	0	0.119249	0.542455	1.45432	1.45432
WS5-1-ファシリティの記録	10	---	---	---	---	---	---	---	---	---	---	---
待ち受け	10	---	---	---	---	---	---	---	---	---	---	---
待ち時間	10	0.344749	0.434152	0.0341752	0.655322	1.25933	0	0	0.0523909	0.706433	0.977661	0.977661
WS5-3-ファシリティの記録	6	---	---	---	---	---	---	---	---	---	---	---
待ち受け	6	---	---	---	---	---	---	---	---	---	---	---
待ち時間	6	0.272166	0.389148	-0.136219	0.680552	1.42982	0	0	0.137757	0.338205	0.998381	0.998381
WorkStation2	36	0.254896	0.498599	0.0861944	0.423597	1.95609	0	0	0	0	2	2
要求待ち要求	36	1.2629	0.78698	0.996624	1.52917	0.623153	0	1	1	2	2	2
バツァ	36	1.2629	0.78698	0.996624	1.52917	0.623153	0	1	1	2	2	2
WorkStation1	56	0.164376	0.453809	0.0428451	0.285907	2.7608	0	0	0	0	2	2
要求待ち要求	56	1.58682	1.15104	1.27857	1.89507	0.725374	0	1	2	3	3	3
バツァ	56	1.58682	1.15104	1.27857	1.89507	0.725374	0	1	2	3	3	3
記録-記録	23	2.94915	0.886102	2.56597	3.33233	0.30046	1.63385	2.59018	2.99213	3.22394	5.06257	3.42872
記録type1-記録	10	---	---	---	---	---	---	---	---	---	---	---
type1	10	1	0	1	1	0	1	1	1	1	1	0
記録type2-記録	10	---	---	---	---	---	---	---	---	---	---	---

図 23.1: サマリ表示

一番左の列は、モデルの出力フォルダにあるデータの一覧が表示されています。構成はメインウィンドウのブラウザパネルと同じです。「+」、「-」を押す事で開閉が出来ます。

列名に表示されるのは、基本統計量であり、以下が表示されます。

- 観測数
- 平均
- 標準偏差
- 分散

- 95 % 信頼区間下限
- 95 % 信頼区間上限
- 変動係数
- 最小値
- 25 % 値
- 中央値
- 75 % 値
- 最大値
- レンジ

モニターか時系列モニターかで統計量の計算方法は異なります。各統計量の計算式は、psim 言語リファレンスマニュアルを参照して下さい。

23.2 データの表示

ブラウザパネルのモニター、時系列モニターあるいは列を右クリックして表示されるメニューから「データを表示する」を選択するとデータを表示する事が出来ます。(図 23.2)

時間	重み	要求待ち	バッファ
1	0.000	0.772	0
2	0.772	1.389	0
3	2.161	0.358	0
4	2.520	0.144	0
5	2.664	0.783	0
6	3.447	3.087	0
7	6.534	0.117	0
8	6.651	0.510	0
9	7.161	0.018	1
10	7.179	0.178	2
11	7.356	0.151	1
12	7.507	0.097	0
13	7.605	0.062	1
14	7.666	0.354	2
15	8.020	0.092	3
16	8.113	0.176	2
17	8.289	0.023	3
18	8.312	0.376	2
19	8.688	0.142	3
20	8.830	0.435	2
21	9.265	0.199	3
22	9.464	0.212	2
23	9.676	1.079	1
24	10.756	0.525	0
25	11.280	0.415	0
26	11.695	0.512	0
27	12.207	0.168	0
28	12.375	0.302	0

図 23.2: データ表示

23.2.1 モニター

モニターを選んでデータを表示した場合、そのモニターに含まれる全ての列が表示されます。

23.2.2 モニターの列

モニターの列を選んでデータを表示した場合、その列のみのデータが表示されます。

23.2.3 時系列モニター

時系列モニターを選んでデータを表示した場合、そのモニターに含まれる全ての列と「時間」という列名で時間列が表示されます。

23.2.4 時系列モニターの列

時系列モニターの列を選んでデータを表示した場合、その列と「時間」という列名で時間列が表示されます。

23.3 データ表示機能

データ表示では、各列の整列や値の検索を行うことができます。複数列をキーにした整列や、前方一致や後方一致などの検索といった高度な設定を行うことが可能です。

23.3.1 整列

データを整列させる事が出来ます。複数キーのソートも可能です。

データを整列させるには、各列の列名の横にあるソートアイコンをクリックするか、右クリックメニューで表示される整列メニューを実行します。

ソートアイコンをクリックすると、「降順」、「昇順」、「レコード順」の順に整列状態が変化します。複数の列の整列状態を変化させることで、複数キーのソートを行う事が出来ます。右クリックメニューの「整列」は、必ず単一キーでのソートになります。右クリックメニューの「複数（複数キー）」の場合、複数キーのソートを行う事が出来ます。

第 23 章 データ表示

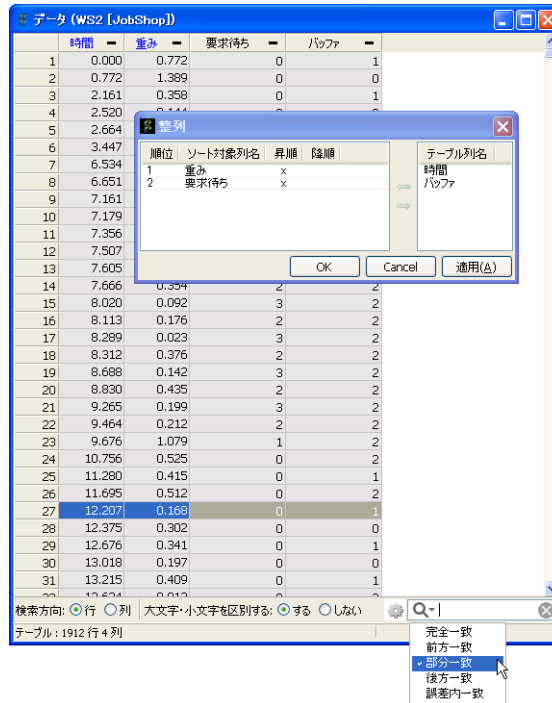


図 23.3: データ表示機能

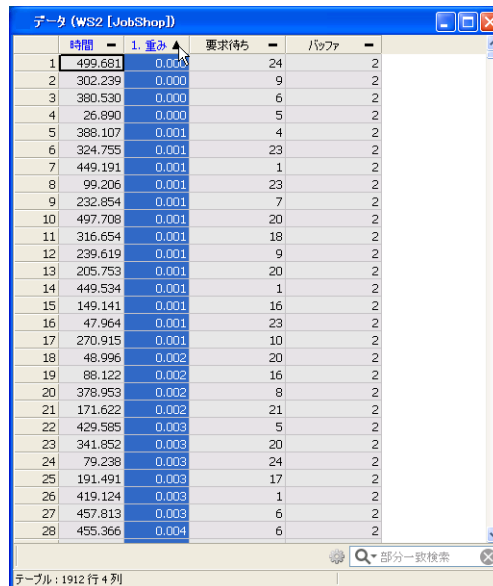


図 23.4: データの整列

23.3 データ表示機能

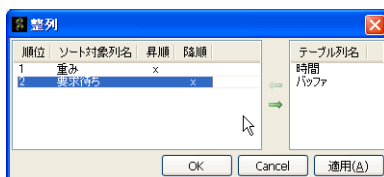


図 23.5: 複数キーのソート

23.3.2 検索

データを検索することが出来ます。完全一致や部分一致などの他に、数値に対しては誤差内にある値を検索（誤差内一致検索）することが出来ます。

データを検索するには、下部右側にある検索フィールドに検索語を入力して Enter キーを押します。続けて Enter キーを押すことで、インクリメンタルに検索を行います。

順位	時間	重み	要求待ち	バッファ
1	0.000	0.772	0	1
2	0.772	1.389	0	0
3	2.161	0.358	0	1
4	2.520	0.144	0	0
5	2.664	0.783	0	1
6	3.447	3.087	0	0
7	6.534	0.117	0	1
8	6.651	0.510	0	2
9	7.161	0.018	1	2
10	7.179	0.178	2	2
11	7.356	0.151	1	2
12	7.507	0.097	0	2
13	7.605	0.062	1	2
14	7.666	0.354	2	2
15	8.020	0.092	3	2
16	8.113	0.176	2	2
17	8.289	0.023	3	2
18	8.312	0.376	2	2
19	8.688	0.142	3	2
20	8.830	0.435	2	2
21	9.265	0.199	3	2
22	9.464	0.212	2	2
23	9.676	1.079	1	2
24	10.756	0.525	0	2
25	11.280	0.415	0	1
26	11.695	0.512	0	2
27	12.207	0.168	0	1
28	12.375	0.302	0	0

図 23.6: データの検索

以下の検索が実行できます。

検索方法	説明
完全一致	検索語と値が一致するものを検索
前方一致	検索語から始まる値を検索
部分一致	検索語を含む値を検索
後方一致	検索語で終わる値を検索
誤差内一致	一定の誤差内にある値を検索

検索フィールドの左側にある虫眼鏡アイコンをクリックし、出てきたメニューで検索方法を選択します。さらに左隣にある歯車アイコンをクリックすると、検索時のオプションを設定できます。オプションでは、検索方向や大文字・小文字の区別、誤差を指定できます。

オプション	説明
検索方向	行毎または列毎のどちらの方向に検索するかを指定
大文字・小文字を区別する	誤差内一致以外の検索の際に大文字・小文字を区別するかしないかを指定
誤差	誤差内一致検索の際の誤差を指定

第24章 データ分析

S⁴ Simulation System ではシミュレーション結果に対する分析機能として、シミュレーションパラメータと結果に対して線形回帰と分散分析を行う分散分析機能がごさいます。

24.1 分散分析

分散分析機能は、ワークスペースタブの出力フォルダ以下にある、モニターまたは時系列モニターのコンテキストメニュー (図 24.1) からアクセスします。

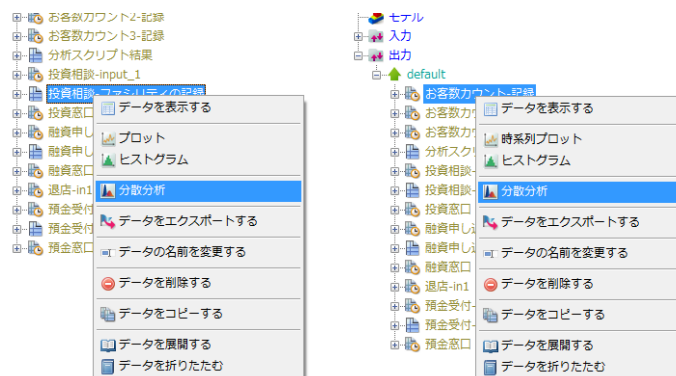


図 24.1: モニター・時系列モニターのコンテキストメニュー

コンテキストメニューの「分散分析」メニューから、分散分析を実行出来ます。分散分析メニューを選択すると、図 24.2 のような分散分析ウィンドウが表示されます。

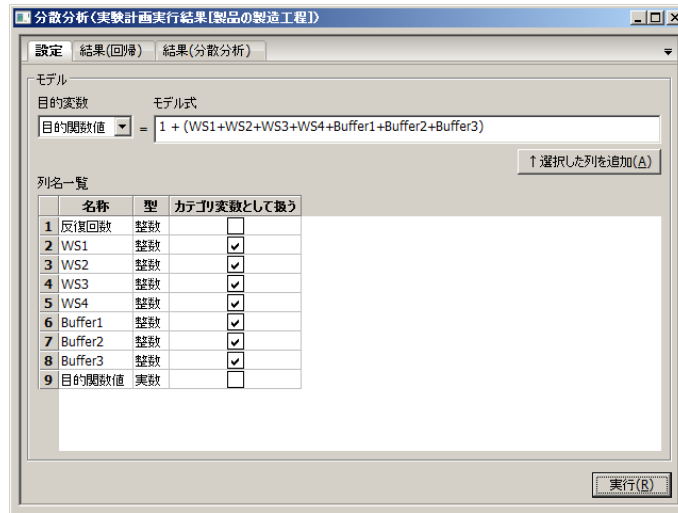


図 24.2: 分散分析ウィンドウ

分散分析ウィンドウには、分散分析の設定を行う「設定」タブと、線形回帰の結果が表示される「結果（回帰）」タブ、分散分析の結果が表示される「結果（分散分析）」タブがあります。

それぞれのタブの構成は以下のようにになっています。

設定タブ 設定タブでは、回帰モデルの設定を行います。

目的変数

モニターから目的変数となる列を選択します。文字列型は選択出来ません。

モデル式

回帰モデルに対するモデル式を入力します。モデル式の設定の仕方は 5.4.5 章の「モデル式」の設定欄と同様です。

↑ 選択した列を追加

列名一覧の表で選択状態となっている列を、説明変数としてモデル式に入れます。

列名一覧

対象のモニター内の列が表示されます。カテゴリ変数として扱う列を選択したり、「↑選択した列を追加」ボタンによってモデル式に追加する列を選択したりします。

表における各列の意味は次の通りです。

名称

列名をあらわします。

型

列の型を表します。

カテゴリ変数として扱う

各列を説明変数として扱う際にカテゴリ変数として扱うか否かを表します。チェックが入っているとカテゴリ変数として扱われ、入っていないと扱われません。

文字列型の列は常にカテゴリ変数として扱われます。

実行

実行ボタンを押すと、設定タブの設定で、線形回帰と分散分析が実行されます。実行結果はそれぞれ「結果（回帰）」タブと「結果（分散分析）」タブに出力されます。

結果（回帰）タブ 「結果（回帰）」タブには、線形回帰の結果が、図 24.3 のような表形式で出力されます。

パラメータ名	係数推定値	標準誤差	t 値	$Pr(> t)$
パラメータ名 ₁	係数 ₁	標準誤差 ₁	t 値 ₁	p_1
⋮	⋮	⋮	⋮	⋮
パラメータ名 _k	係数 _k	標準誤差 _k	t 値 _k	p_k

図 24.3: 結果（回帰）

各列の意味は次の通りです。

パラメータ名 回帰モデル中の各説明変数の名称です。カテゴリ変数の列に対しては”列名 [T. カテゴリ値]”の形で各カテゴリ値に対応する説明変数があらわされます。

係数推定値 説明変数に対応する係数です。

標準誤差 係数の標準誤差です。

t 値 係数の t 値です。

$Pr(> |t|)$ 係数の p 値です。

結果（分散分析）タブ 「結果（分散分析）」タブには、分散分析の結果が、[図 24.4](#) のような表形式で出力されます。

パラメータ名	平方和	自由度	平均平方	F 値	$Pr(> F)$
パラメータ名 ₁	平方和 ₁	自由度 ₁	平均平方 ₁	F 値 ₁	p_1
⋮	⋮	⋮	⋮	⋮	⋮
パラメータ名 _k	平方和 _k	自由度 _k	平均平方 _k	F 値 _k	p_k

図 24.4: 結果（分散分析）

各列の意味は次の通りです。

パラメータ名 回帰モデル中の各説明変数の名称です。

平方和 説明変数に対応するタイプ III の平方和です。

自由度 説明変数の対応する自由度です。

平均平方 説明変数の対応する平均平方です。

F 値 説明変数に対応する F 値です。

$Pr(> |t|)$ 説明変数に対応する p 値です。

第25章 強化学習結果の閲覧

強化学習モデルアイコン (14章) を使用したシミュレーションを実行すると、実行結果として強化学習の学習結果が出力されます¹。強化学習の学習結果は強化学習モデルビューアを通して閲覧できます。

25.1 強化学習モデルビューア

強化学習モデルビューアは、ワークスペースタブの入力または出力フォルダ以下にある、強化学習モデルのコンテキストメニューからアクセスします。

コンテキストメニューの「学習モデルを閲覧する」メニューから強化学習モデルビューアを起動します。「学習モデルを閲覧する」メニューを選択すると、図 25.1 のようなウィンドウが表示されます。



図 25.1: 強化学習モデルビューア

¹設定により出力させないことも出来ます。

強化学習モデルビューアのウィンドウには、閲覧対象の選択などの各種設定を行う「設定」タブと、設定に従って学習結果（の一部）が表示される「結果」タブがあります。強化学習の学習結果は一般には表形式では表現できず、様々な形式で保存されます。そのため、「設定」タブで学習結果のうち一部をサンプリングする設定を行い、サンプリングした結果を表形式で「結果」タブに出力するような形になっています。

それぞれのタブの構成は以下のようにになっています。

25.1.1 設定タブ

閲覧対象の選択や、閲覧範囲の設定を行います。タブ内には、設定項目と「計算」ボタンがあります。設定を行った後、「計算」ボタンをクリックすると、設定に従って学習結果の一部が抽出され「結果」タブに出力されます。

以下の設定項目を持ちます。

強化学習モデル

対象となる強化学習モデルをあらわします。

ID

エージェントシミュレーションで強化学習した際など、場合によっては対象の学習結果には複数の学習結果が格納される場合があります（エージェントの場合はエージェントごとの学習結果になります）。複数の学習結果が格納されている場合は、いずれの学習結果であるかを選択します。複数の結果を保持していない場合は、選択できません。

閲覧対象

閲覧対象の学習結果を選択します。

観測値の範囲

閲覧対象の観測値の範囲を指定します。観測値集合の次元ごとに取りうる値を指定し、それらの直積集合を範囲として指定できます。

行動値の範囲

閲覧対象の行動値の範囲を指定します。行動値集合の次元ごとに取りうる値を指定し、それらの直積集合を範囲として指定できます。

25.1.2 結果タブ

設定タブで設定を行った後、計算ボタンをクリックすると設定に従って学習結果の一部が抽出され「結果」タブに出力されます。

結果タブに出力された結果が出力は、モニタとして保存する事ができます。

結果タブ内にある「モニタとして出力」ボタンをクリックすると図 25.2 のように保存するモニタの名前を訊かれます。名前を入力すると、入力された名前で結果がモニタとして出力フォルダに出力されます。

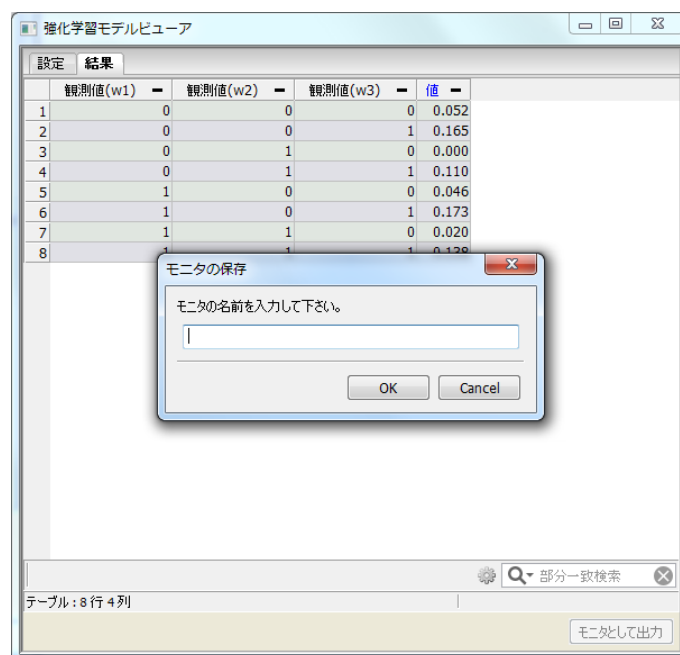


図 25.2: 強化学習モデルビューアの結果をモニタとして出力する

第26章 アニメーション

アニメーションとは、待ち行列の状態をリアルタイムにモデル上に可視化する機能です。この機能を用いる事でシミュレーション実行中にどこに待ち行列が発生しやすいかを視覚的に確認することができます。アニメーションを用いるには、アニメーション全体の動作に関する設定をパラメータ編集画面で、個々の部品の動作に関する設定を部品編集画面で設定する必要があります。

26.1 パラメータ設定

アニメーション全体の動作に関する設定を、パラメータ編集画面の基本設定タブにあるアニメーション欄で設定します。具体的なパラメータは以下の通りです。

表示

アニメーションを表示するかどうかの設定をします。

表示間隔

アニメーションを表示する間隔を設定します。ここで設定する値は、CPU 時間ではなくシミュレーションモデルでの時間です。

タイプ

アニメーションの表示方法を設定します。表示方法は、「表示タイプ」、「色」、「表示範囲」の3つのパラメータの組み合わせで設定します。各パラメータの選択項目は以下の通りです。

表示タイプ	説明
bar	状態を長方形の長さで表示します
circle	状態を円で表示します

26.1 パラメータ設定

色	説明
グラデーション	表示タイプで指定した形がグラデーションで変化します
3色	表示タイプで指定した形が3色のパターンで変化します (表示タイプに circle を指定した時のみ設定可)

表示範囲	説明
ユーザ指定	部品パラメータでユーザが指定した色に従います。
指定なし	部品パラメータで選択したグラデーションパターンを使用します (色にグラデーションを指定した時のみ設定可)

表示内容

待ち行列の長さの表示内容を選択します。

表示内容	説明
数	待ち行列の長さの数を表示します
割合	待ち行列の長さのキューサイズに占める割合を表示します (キューサイズが無制限の場合には、部品パラメータの表示範囲の幅がキューサイズになります)



図 26.1: アニメーションのパラメータ

26.2 部品パラメータ設定

個々の部品編集画面でアニメーション表示の為の条件や範囲を指定します。編集画面はタイプがグラデーションと 3 色の場合で異なります。タイプにグラデーションを指定した場合はアニメーション表示範囲欄で以下の項目を設定します。

単位

アニメーションで表示する範囲をキューサイズ (同時利用容量) の割合で指定するか、数値で指定するかを設定します。

範囲

アニメーションで表示する範囲の上下限值で指定します (キューのサイズが 100 で表示範囲の上下限が 0 と 10 の場合には、待ち行列の長さが 0 の時に 0 % 待ち行列の長さが 10 以上の時に 100 % となります)。

ユーザ指定

グラデーションの下限色と上限色を設定します。

指定なし

グラデーションパターンを選択します。

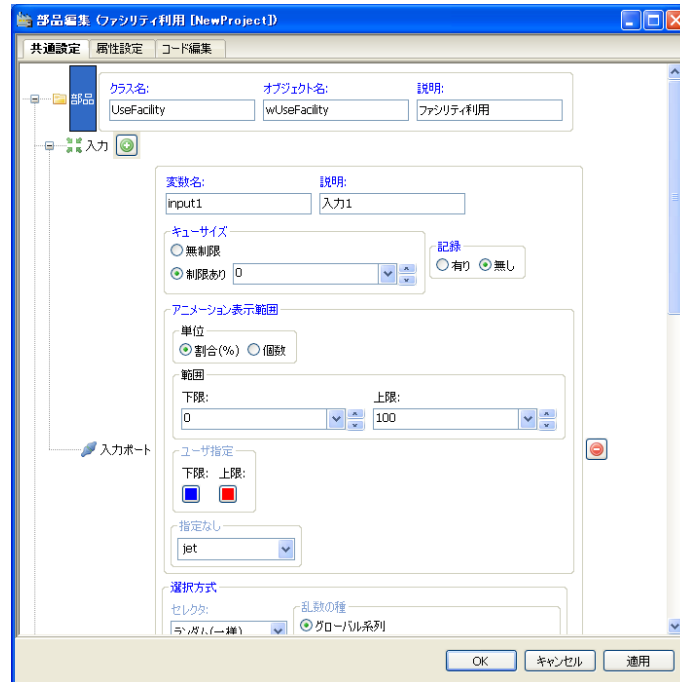


図 26.2: グラデーションの編集画面

タイプに 3 色を指定した場合は、以下の項目を設定します。

条件 1

条件 1 の個数と色を設定します。

条件 1 の個数 \leq 待ち行列の長さ $<$ 条件 2 の個数の場合にここで指定した色になります。

条件 2

条件 2 の個数と色を設定します。

条件 2 の個数 \leq 待ち行列の長さ $<$ 条件 3 の個数の場合にここで指定した色になります。

条件 3

条件 3 の個数と色を設定します。

条件 3 の個数 \leq 待ち行列の長さ

の場合にここで指定した色になります。

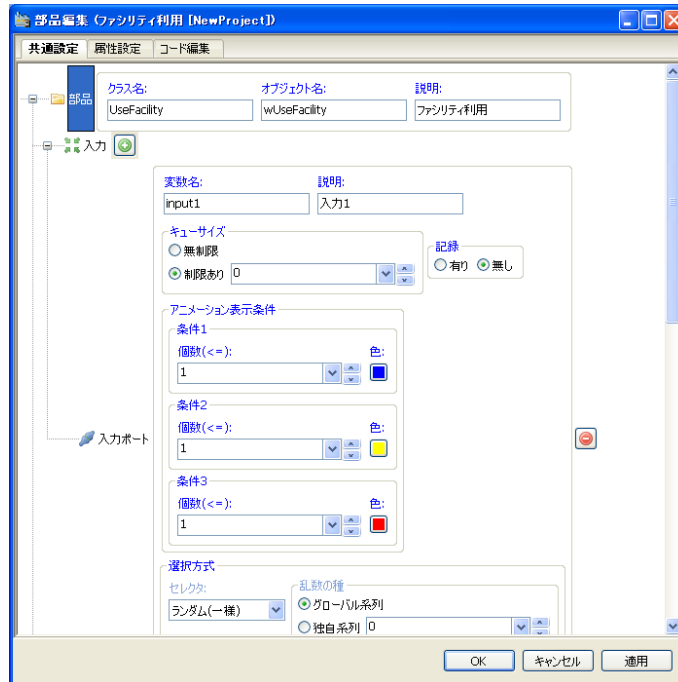


図 26.3: 3 色の部品編集画面

第27章 VAP 連携

S⁴ Simulation System をインストールした PC に VAP (isual Analytics Platform) がインストールされている場合、VAP 上から S⁴ Simulation System を呼び出す事が出来ます。

具体的には、以下のような事が出来ます。

- VAP 上で作成したパラメータを S⁴ Simulation System に渡してシミュレーションを行う事が出来る。
- S⁴ Simulation System 上で行ったシミュレーションを VAP へ返却し、VAP 上で、分析を行う事が出来る。

27.1 操作

27.1.1 配置

VAP (Visual Analytics Platform) がインストールされている PC 上に、S⁴ Simulation System をインストールすると、VAP のオブジェクトブラウザ内の、アドオンフォルダに、「S-Quattro Simulation System」が表示されます。更にその下に、「Simulation」アドオンが表示されます。(図 27.1)

もし、アドオンが表示されていない場合は、S⁴ Simulation System をインストール後に、VAP をインストールした場合などが考えられます。その場合は、S⁴ Simulation System を再度インストールして下さい。

VAP の通常の操作通りに、Simulation アドオンをプロジェクト上に配置する事が出来ます。

S-Quattro 上のシミュレーションモデル内で利用したいデータがある場合は、Simulation アドオンに対し、任意個のリンクを結ぶ事が出来ます。

S-Quattro 上のシミュレーションモデル上の結果を VAP に返却したい場合は、Simulation アドオンから、任意個のリンクを結ぶ事が出来ます。

27.1.2 実行

Simulation アドオンをダブルクリック、あるいは右クリックで表示されたメニューから「実行」を選択すると、S⁴ Simulation System が VAP アドオンモードで起動します。(図 27.2)

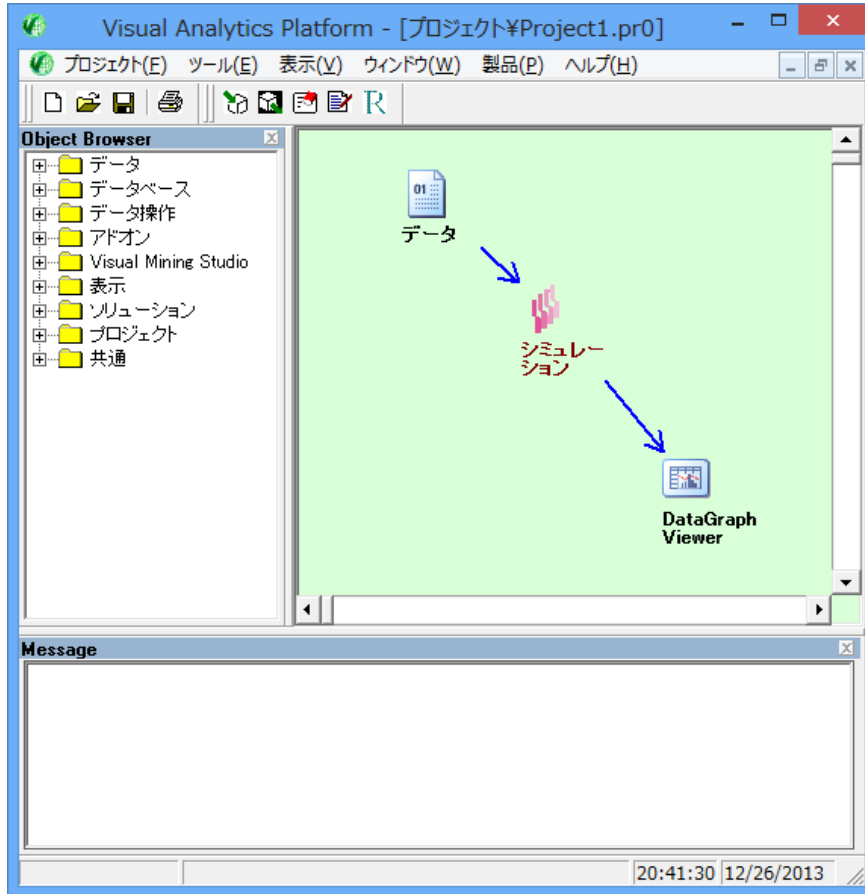


図 27.1: VAP S-Quattro アドオン

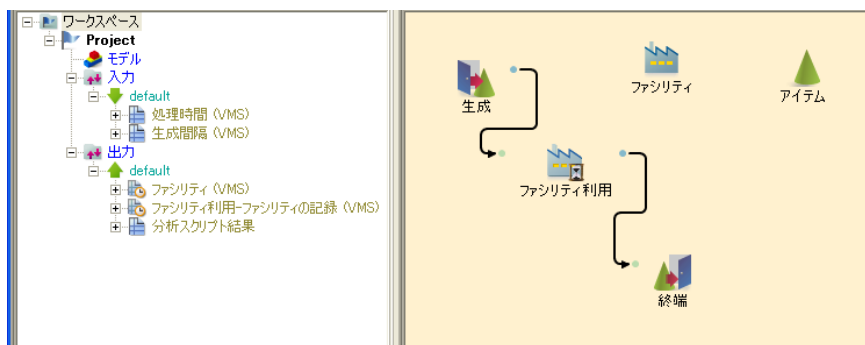


図 27.2: VAP S-Quattro アドオン起動画面

VAP アドオンモードは、基本的な操作方法は、スタンドアロンで起動した時と同じですが、以下の点で異なります。

- 独自のワークスペースを持ち、プロジェクト名は必ず「Project」になります。
- 入力フォルダ内には、VAP で指定した入力リンクのデータがインポートされており、名前の末尾に (VAP) が付与されています。新規に入力データを追加したい場合は、入力フォルダ内にデータを用意し、モニター上で右クリックを押し、表示されたメニューの中の「VAP から入力する」にチェックを入れて下さい。次回起動時に入力マッチングウィンドウが表示されます。入力データを削除したい場合は、モニター上で右クリックを押し、表示されたメニューの中の「VAP から入力する」のチェックを外して下さい。次回起動時に入力マッチングウィンドウが表示されます。
- 出力フォルダ内には、シミュレーションの実行結果が保存されますが、VAP 側に出力するデータには名前の末尾に、(VAP) が付与されています。新規に出力データを追加したい場合は、モニター上で右クリックを押し、表示されたメニューの中の「VAP へ出力する」にチェックを入れて下さい。出力データを削除したい場合は、モニター上で右クリックを押し、表示されたメニューの中の「VAP へ出力する」のチェックを外して下さい。

27.1.3 再実行

Simulation アドオンより下のアイコンをダブルクリック、あるいは Simulation アドオンで右クリックで表示されたメニューから「再実行」を選択すると、S⁴ Simulation System がバックグラウンドで実行され、S⁴ Simulation System の進捗ダイアログのみが表示されます。

27.1.4 入出力マッチング

S-Quattro アドオン上で、「VMS から入力する」と設定したデータ数を変化させた場合、次回に S-Quattro アドオンの起動時に入出力マッチングウィンドウが表示されます。適宜入出力マッチングを設定して下さい。

27.1.5 実行結果

S-Quattro アドオン上で、「VMS へ出力する」と設定したデータは VAP 側に返却されます。VAP 上に返却されたデータは通常の VAP と同様の処理

を行う事が出来ます。複数のデータが返されますので、適宜入力マッチングを行う必要があります。また、第 1 テーブルには必ず「result..」という名前のデータが返却されます。

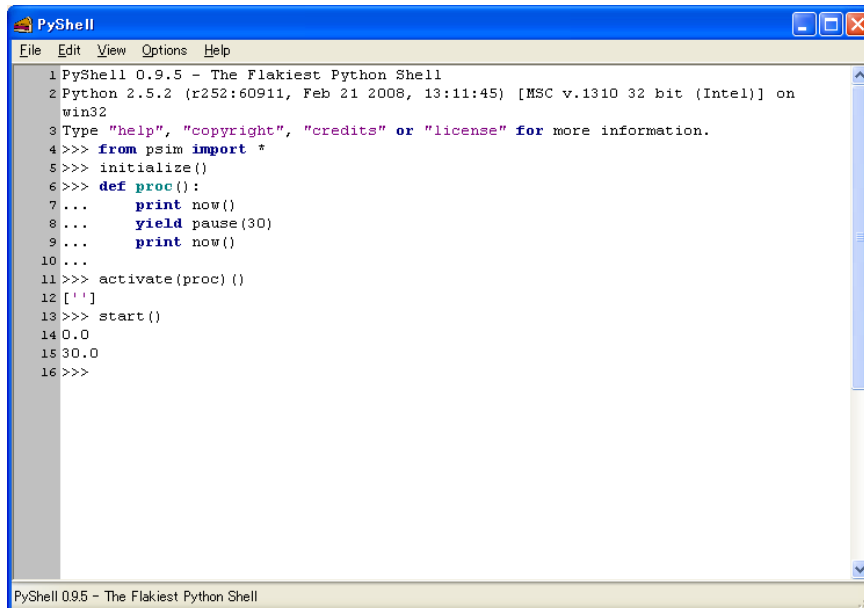
第28章 その他

28.1 Python シェルの起動

Python シェルについてを説明します。

ウィンドウメニュー (3.2.4 章) の「Python シェル起動」を選択すると、Python シェルが起動されます。

Python シェルでは Python のプロンプトが表示されますので、Python の動作を確認する事が出来ます。psim 言語もインポートされた状態で起動します。(図 28.1)



```
1 PyShell 0.9.5 - The Flakiest Python Shell
2 Python 2.5.2 (r252:60911, Feb 21 2008, 13:11:45) [MSC v.1310 32 bit (Intel)] on
  win32
3 Type "help", "copyright", "credits" or "license" for more information.
4 >>> from psim import *
5 >>> initialize()
6 >>> def proc():
7 ...     print now()
8 ...     yield pause(30)
9 ...     print now()
10 ...
11 >>> activate(proc) ()
12 ['']
13 >>> start()
14 0.0
15 30.0
16 >>>
```

図 28.1: Python シェル

関連図書

- [1] Geraerts R.J. and Overmars M.H. 'The Corridor Map Method: A General Framework for Real-Time High-Quality Path Planning' Computer Animation and Virtual Worlds, volume 18, pp. 107 - 119. 2007.
- [2] D. Helbing and P. Molnar, 'Social Force Model for Pedestrian Dynamics', Physical Review E, 51, 5 1995.
- [3] H. R. Maei and R. S. Sutton, 'GQ(λ): A general gradient algorithm for temporal-difference prediction learning with eligibility traces,' 2010.
- [4] W. T. Miller and F. H. Glanz, 'UNH.CMAC Version 2.1. The University of New Hampshire Implementation of the Cerebellar Model Arithmetic Computer - CMAC,' 1994:1-19, 1996.
- [5] R. S. Sutton and A. G. Barto, 'Reinforcement Learning: An Introduction,' MIT Press, Cambridge, MA, 1998, A Bradford book.
- [6] 北川源四郎, 'モンテカルロ・フィルタおよび平滑化について,' 統計数理, 1996, 44(1), 31-48.

索引

3D アニメーション

3D アニメーション, [258](#)

エージェント

NW エージェント, [191](#)

SFM エージェント, [187](#)

同期エージェント, [184](#)

非同期エージェント, [193](#)

粒子フィルタエージェント, [189](#)

グラフ

グラフ機能, [309](#)

フローアイテム

アイテム, [179](#)

オペレーター, [271](#)

お客, [278](#)

問い合わせ, [272](#)

環境

BarabasiAlbert グラフ, [217](#)

GEXF フォーマットグラフ, [209](#)

GNM グラフ, [207](#)

GNP グラフ, [205](#)

GraphML フォーマットグラフ,
[204](#)

NW 地図, [201](#)

PowerlawCluster グラフ, [199](#)

SFM 環境, [197](#)

SFM 地図, [196](#)

カスタムグラフ, [214](#)

ユークリッド 2D, [211](#)

格子グラフ, [202](#)

完全グラフ, [215](#)

環境, [212](#)

強化学習モデル

エージェント用強化学習モデル,
[242](#)

離散イベント用強化学習モデル,
[240](#)

資源

イベント, [172](#)

ストア, [174](#)

タンク, [175](#)

ファシリティ, [173](#)

窓口, [277](#)

対応待ちリスト, [271](#)

部品

イベント待ち, [138](#)

イベント発火, [139](#)

ストアから取得, [142](#)

ストアへ追加, [144](#)

タンクから取得, [146](#)

タンクへ追加, [147](#)

ファシリティ解放, [141](#)

ファシリティ要求, [139](#)

ファシリティ利用, [141](#)

マージ, [150](#)

記録, [156](#)

強化学習設定, [160](#)

結合, [149](#)

時間確認, [273](#)

受電, [274](#)

終端, [137](#)

出社, [276](#)

生成, [136](#)

設定, [151](#)

窓口選択部品：レジ, 280
窓口選択部品：切符購入窓口,
280
窓口到着, 279
窓口利用部品：レジ, 281
窓口利用部品：切符購入窓口,
281
即時報酬確定, 159
対応, 275
対応開始, 275
対応終了, 273, 280
対応待ち, 275
退社, 276
遅延, 153
通過, 156
入電, 274
判断, 152
複製, 155
分割, 151
放棄呼, 273
連続, 157

変数

補助変数, 182
連続変数, 181

0