



SCENARGIE®

Scenargie®2.1 Multi-Agent Extension Module

ユーザガイド

Space-Time Engineering, LLC

2016 年 9 月

目次

はじめに.....	1
1. プロダクトの内容.....	2
2. インストール方法.....	3
2.1. Linux、MacOS 環境へのインストール.....	3
2.1.1. シミュレータ実行ファイルの作成.....	3
2.1.2. Visual Lab で使用する場合の設定 Linux 編.....	5
2.2. Windows 環境へのインストール.....	6
2.2.1. シミュレータ実行ファイルの作成.....	6
2.2.2. Visual Lab で使用する場合の設定.....	7
2.3. サンプルシナリオ.....	9
2.3.1. multiagent_commuter.....	9
2.4. 旧サンプルシナリオの実行.....	11
2.4.1. Scenargie1.8 r19737 版/Scenargie2.0 r19737 版サンプルシナリオの実行.....	11
2.4.2. Scenargie2.1 r20324 版サンプルシナリオの実行.....	11
3. エージェント定義ファイル.....	12
3.1. ファイル構文.....	12
3.2. エージェントプロファイル定義ファイル.....	12
3.3. エージェント行動設定ファイル.....	16
3.4. エージェントタイムテーブル設定ファイル.....	19
4. プロパティ.....	20
4.1. プロパティ一覧.....	20
5. 統計値、トレース.....	23
5.1. 統計値の標準設定一覧表.....	23
5.2. トレースの標準設定一覧表.....	23
6. 参考文献.....	24


はじめに

本書は、離散事象シミュレータ Scenargie2.1 Multi-Agent Extension Module の操作方法を示すものです。

関連ドキュメント

インストレーションガイド
プログラマーズガイド
Visual Lab ユーザガイド
Base Simulator ユーザガイド
Multi-Agent Extension Module モデルリファレンス

1. プロダクトの内容

Scenargie2.1 Multi-Agent Extension Module は、Scenargie2.1 Base Simulator 用オプションモジュールで、Base Simulator 本体と組み合わせて使用することを想定しています。(図 1-1  に示す部分)

Scenargie2.1 Multi-Agent Extension Module には以下の内容が含まれます。

- Intelligent Driver Model [1]
- マルチエージェントシミュレーションによるモビリティモデル (歩行、自転車、自動車、タクシー、バス、電車)
- makefile
- サンプルシナリオ

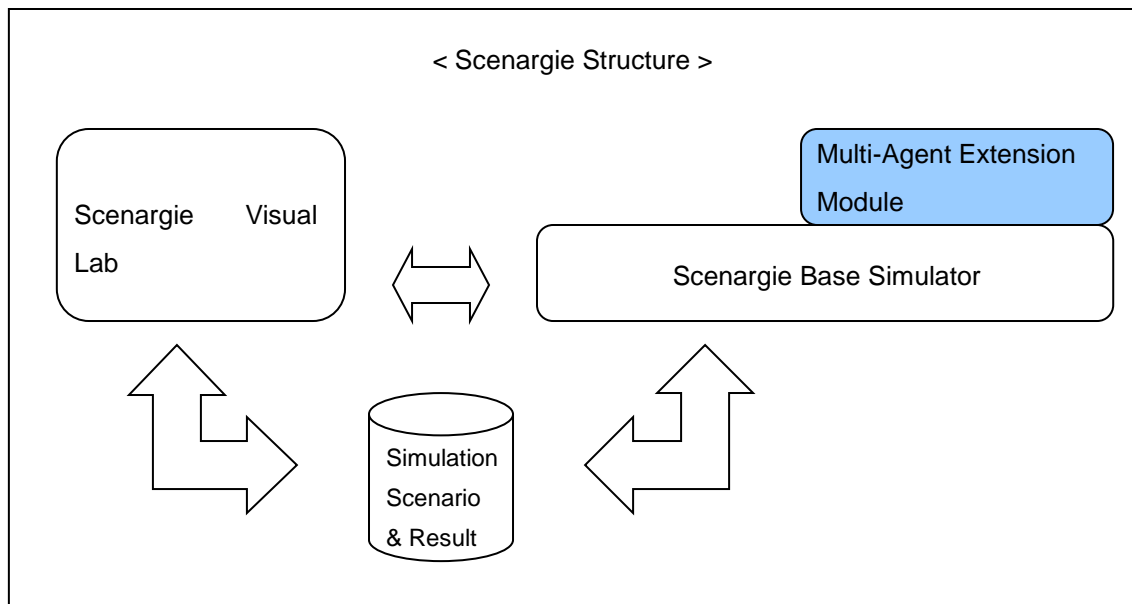


図 1-1 Scenargie システム構成

2. インストール方法

Scenargie Multi-Agent Extension Module パッケージは、Linux 環境、MacOS 環境および Windows 環境へインストールが可能です。Scenargie Base Simulator のインストール後に Scenargie Multi-Agent Extension をインストールしてください。

Scenargie Base Simulator および Scenargie Multi-Agent Extension Module インストール後のソースコードディレクトリ構成は下記ようになります。

```
scenargie_simulator/2.1
|-- document
|-- package_tree
|-- scenarios_linux
|-- scenarios_windows
`-- source
    |-- base
    |-- boost
    |-- multiagent
    |-- include
    |-- multisystems
    |-- objlibs
    |-- `-- simulator
|-- util
`--visuallab
    |-- data
    `-- sample
```

2.1. Linux、MacOS 環境へのインストール

2.1.1.シミュレータ実行ファイルの作成

1) パッケージの展開

以下の説明では、Scenargie Base Simulator がユーザホームディレクトリ下にディレクトリ「scenargie_simulator」として展開されているものとします。Scenargie Multi-Agent Extension Module のパッケージをユーザホームディレクトリ下にコピーまたは移動し、パッケージを展開してください。

<コマンド例>

```
$ cp Scenargie-2.1-MultiAgentExtensionModule-rxxxx.zip ~/
$ cd ~/
$ unzip Scenargie-2.1-MultiAgentExtensionModule-rxxxx.zip
```

2) 実行ファイルの作成

Scenargie Multi-Agent Extension Module のインストールにより `scenargie_simulator/2.1/source/multiagent/` が作成され、このディレクトリにサンプルメイクファイル `makefile.linux` が含まれています。このサンプルメイクファイルを使用して下記のように実行ファイルを作成します。

<コマンド例>

```
$ cd ~/scenargie_simulator/2.1/source/multiagent
$ make -f makefile.linux
```

上記ビルドが正常に終了した後、カレントディレクトリにシミュレータ実行ファイル「sim」が作成されます。

ビルド時にオプションを指定することで、並列ビルドや IPv6 の有効化などが行えます。また、複数のオプションモジュールを組み合わせる場合は、マルチシステム用の `makefile` を使用してビルドします。その際、ビルドオプションとして有効化するオプションモジュールを指定します。詳しくは、「インストレーションガイド」を参照ください。

注意)

一度シミュレータ実行ファイルの作成を行った後、再作成する場合、シミュレータ実行ファイルやオブジェクトファイルを事前に削除することを推奨します。

<コマンド例>

```
$ cd ~/scenargie_simulator/2.1/source/multiagent
$ make -f makefile.linux clean
$ make -f makefile.linux
```

2.1.2. Visual Lab で使用する場合の設定 Linux 編

Linux 環境で Scenargie Visual Lab で使用する場合、以下の項目も設定します。

1) データファイルのコピー

Scenargie Multi-Agent Extension Module パッケージには Scenargie Visual Lab で Multi-Agent の機能を使用するためのデータファイルが含まれます。このデータファイルを Scenargie Visual Lab のインストールディレクトリにコピーします。

データファイル:

```
scenargie_simulator/2.1/visuallab/data/multiagent/multiagent.component
scenargie_simulator/2.1/visuallab/data/multiagent/multiagent.objtype
```

コピー先ディレクトリ:

Scenargie Visual Lab インストールディレクトリ下の「data/multiagent/」ディレクトリ下
例) visuallab/data/multiagent/

2) シミュレータ実行ファイルの指定

[Tools]- [Object Properties] Global: Simulation Executable Name に作成したシミュレータ実行ファイルを指定します。なお、シナリオ新規作成時のシミュレータ実行ファイルの初期値を変更する事も可能です。「Scenargie Visual Lab ユーザガイド」オブジェクトタイプ編集機能を参照の上、Property: Executable Name の初期値を変更してください。

2.1.3. Visual Lab で使用する場合の設定 MacOS 編

MacOS 環境で Scenargie Visual Lab で使用する場合、以下の項目も設定します。

1) データファイルのコピー

Scenargie Multi-Agent Extension Module パッケージには Scenargie Visual Lab で Multi-Agent の機能を使用するためのデータファイルが含まれます。このデータファイルが含まれるディレクトリ (data/multiagent) を Scenargie Visual Lab のインストールディレクトリにコピーします。

データファイル:

```
scenargie_simulator/2.1/visuallab/data/multiagent/multiagent.component
scenargie_simulator/2.1/visuallab/data/multiagent/multiagent.objtype
```

コピー先ディレクトリ:

Scenargie Visual Lab インストールディレクトリ下の「data/」ディレクトリ下
例) /visuallab/data/multiagent/

2) シミュレータ実行ファイルの指定

[Tools]- [Object Properties] Global: Simulation Executable Name に作成したシミュレータ実行ファイルを指定します。

2.2. Windows 環境へのインストール

Scenargie Multi-Agent Extension Module の Windows 環境へのインストールは管理者アカウントで実施します。

<コマンド例>は、Visual C++のインストールにより作成される「Visual Studio 2013 Command Prompt」での実行を想定しています。

32bit でのビルドを行う場合は、

[スタート]-[プログラム]-[Visual Studio 2013]-[Visual Studio Tools]-[VS2013 x86 Native Tools Command Prompt]をクリックして起動される Command Prompt を使用してください。

64bit でのビルドを行う場合は、

[スタート]-[プログラム]-[Visual Studio 2013]-[Visual Studio Tools]-[VS2013 x64 Cross Tools Command Prompt]をクリックして起動される Command Prompt を使用してください。

2.2.1.シミュレータ実行ファイルの作成

1) パッケージの展開

以下の説明では、Scenargie Base Simulator がCドライブのルート下に「C:\scenargie_simulator」として展開されているものとします。Scenargie Multi-Agent Extension Module のパッケージを C ドライブのルート下にコピーまたは移動し、パッケージを展開してください。

<操作例>

エクスプローラ等で入手したパッケージ Scenargie-2.1-MultiAgentExtensionModule-rxxxx.zip を C:\¥に展開する。

2) makefile の編集と実行ファイルの作成

Scenargie Multi-Agent Extension Module のインストールにより scenargie_simulator¥2.1¥source¥multiagent¥ が作成され、このディレクトリにサンプルメイクファイル makefile.win が含まれています。このサンプルメイクファイルを使用して下記のように実行ファイルを作成します。

<コマンド例>


```
> cd C:\scenargie_simulator¥2.1¥source¥multiagent
> nmake -f makefile.win
```

上記ビルドが正常に終了した後、カレントディレクトリにシミュレータ実行ファイル「sim.exe」が作成されます。

ビルド時にオプションを指定することで、並列ビルドや IPv6 の有効化などが行えます。また、複数のオプションモジュールを組み合わせて使用する場合は、マルチシステム用の makefile を使用してビルドします。その際、ビルドオプションとして有効化するオプションモジュールを指定します。詳しくは、「インストレーションガイド」を参照ください。

注意)

一度シミュレータ実行ファイルの作成を行った後、再作成する場合、シミュレータ実行ファイルやオブジェクトファイルを事前に削除することを推奨します。

<コマンド例>

```
> cd C:\scenargie_simulator¥2.1¥source¥multiagent
> nmake -f makefile.win clean
> nmake -f makefile.win
```

2.2.2. Visual Lab で使用する場合の設定

Scenargie Visual Lab で使用する場合、以下の項目も設定します。

1) データファイルのコピー

Scenargie Multi-Agent Extension Module パッケージには Scenargie Visual Lab で Multi-Agent の機能を使用するためのデータファイルが含まれます。このデータファイルが含まれるディレクトリ (data¥multiagent) を Scenargie Visual Lab のインストールディレクトリにコピーします。

データファイル:

```
scenargie_simulator¥2.1¥visuallab¥data¥multiagent¥multiagent.component
scenargie_simulator¥2.1¥visuallab¥data¥multiagent¥multiagent.objtype
```

コピー先ディレクトリ:

```
Scenargie Visual Lab インストールディレクトリ下の「data¥」ディレクトリ下
例) C:\Scenargie¥data¥multiagent¥
```

2) シミュレータ実行ファイルの指定

[Tools]- [Object Properties] Global: Simulation Executable Name に作成したシミュレータ実行ファイルを指定します。

上記 1)および 2)の操作は、インストールスクリプトにより実行可能です。

インストールスクリプト:

```
scenargie_simulator¥2.1¥visuallab¥install-multiagent-win.vbs
```

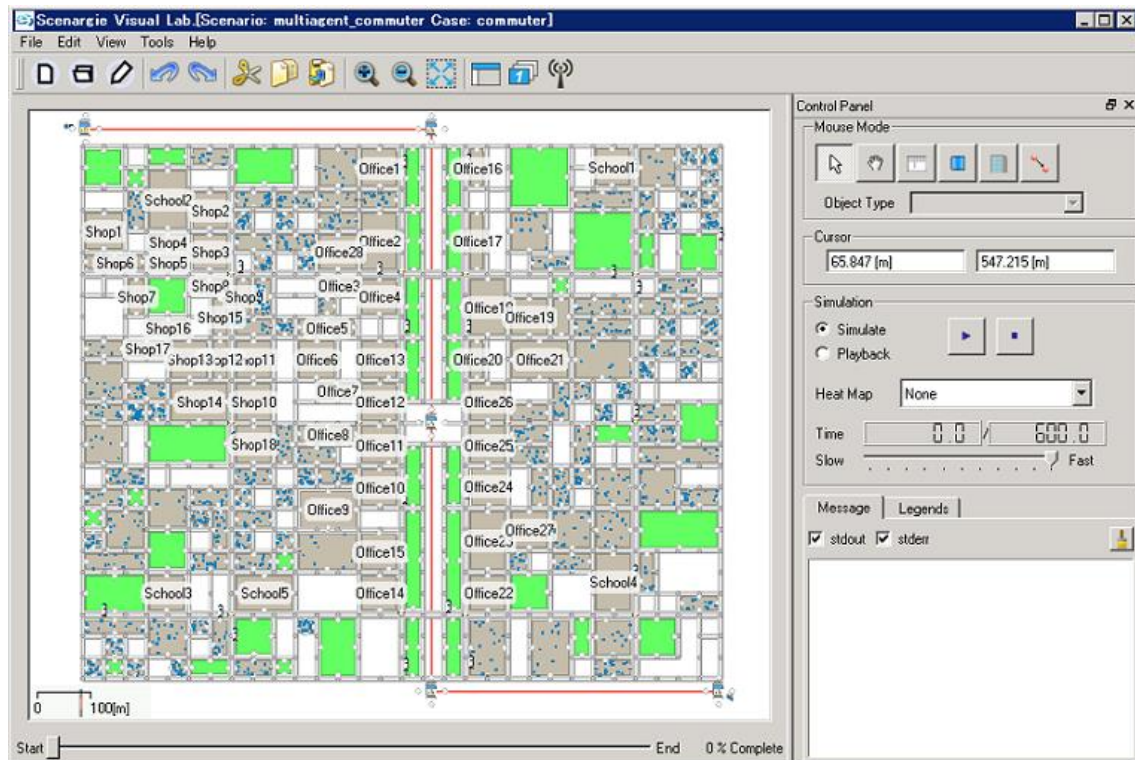
2.3. サンプルシナリオ

サンプルシナリオは、前節に記載の「パッケージの展開」により以下のディレクトリに展開されます。

```
scenargie_simulator/2.1/scenarios_linux/
scenargie_simulator/2.1/scenarios_windows/
```

Multi-Agent Extension Module のサンプルシナリオは multiagent_ をプリフィックスとするディレクトリになります。Scenargie Visual Lab 用のシナリオは当該ディレクトリの .case ファイルを読み込んで使用します。コマンドライン実行用のシナリオは当該ディレクトリ内の commandline ディレクトリに展開されます。

2.3.1. multiagent_commuter



- シナリオ構成

Agent:

- Human × 2000
- Train × 35
- Bus × 102
- Taxi × 70
- Private Car × 160

建物:

- House × 108
- Stationbuilding × 2
- Office × 28
- School × 5
- Shop × 18
- Park × 43

目的地:

Student : School、Park

OfficeWorker : Office、Shop

- シナリオ概要

本シナリオは、Human が House1～108、Stationbuilding1～2 のいずれかを出発地点とし、徒歩を含む交通機関(バス、鉄道、タクシー、自家用車)を利用して、それぞれの目的地へ移動し、また House1～108、Stationbuilding1～2 に戻ります。

2.4. 旧サンプルシナリオの実行

新たにインストールされた Scenargie2.1 Visual Lab、または、新たにビルドされたシミュレータ実行ファイルを用いて旧サンプルシナリオを実行する方法を説明します。

注意) データファイル、サンプルファイルも正しくインストールされていることを確認してください。

2.4.1.Scenargie1.8 r19737 版/Scenargie2.0 r19737 版サンプルシナリオの実行

Visual Lab 用サンプルシナリオの実行

旧サンプルシナリオは、そのまま実行可能です。

コマンドライン用サンプルシナリオの実行

旧サンプルシナリオは、そのまま実行可能です。

2.4.2.Scenargie2.1 r20324 版サンプルシナリオの実行

Visual Lab 用サンプルシナリオの実行

旧サンプルシナリオは、そのまま実行可能です。

コマンドライン用サンプルシナリオの実行

旧サンプルシナリオは、そのまま実行可能です。

3. エージェント定義ファイル

エージェント定義ファイルはエージェントのプロフィール、行動及び交通機関のタイムテーブルを定義するもので、以下のファイルがあります。詳細は「Scenargie Multi-Agent Extension Module モデルリファレンス」を参照してください。

- エージェントプロファイル定義ファイル
- エージェント行動設定ファイル
- エージェントタイムテーブル定義ファイル

3.1. ファイル構文

エージェント定義ファイル共通のファイル構文は以下の通りです。

- 1 項目 1 行とします
- 「#」で始まる行はコメント行を示します
- 改行コードはシミュレーション実行環境に合わせます
- アスキーコードのみを使用します (Station 名等の固有名詞を除く)

3.2. エージェントプロファイル定義ファイル

エージェントのプロフィールを定義します。

「ProfileType」タグでプロファイルタイプ名を定義します。

ProfileType : <プロファイル名>

ある「ProfileType」タグから次の「ProfileType」タグ、またはファイル終了までのパラメータ設定を当該プロファイルタイプのプロファイル定義とします。

Taxi、及び Bus は予約のプロファイルタイプで、エージェントプロファイル定義に必ず記述します。

プロファイルを定義するパラメーター一覧

パラメータ名	値	補足	設定省略時の初期値
Age	実数	年齢	0
WalkSpeed	実数	歩行速度[m/s]	0
BicycleSpeed	実数	自転車速度[m/s]	0
VehicleSpeed	実数	車両の最大速度[m/s]	60 / 3.6
Utility1	実数	ユーティリティ 1	0
Utility2	実数	ユーティリティ 2	0
RecalcIntervalForLastViaPointDelay	実数	経路再計算トリガ(最後に訪れた via ポイントでの遅延に対するトリガ)	60*10

RecalcIntervalForNextViaPointDelay	実数	経路再計算トリガ(次の via ポイントに対しての遅延に対するトリガ)	60*10
RecalcIntervalForDestinationDelay	実数	経路再計算トリガ(目的地への到着予定時刻に対する現在の経路の遅延に対するトリガ)	60*10
RecalcIntervalForVehicleDelay	実数	経路再計算トリガ(乗車予定の路線の遅れに対するトリガ)	60*10
RecalcThresholdForCongestion	実数	経路再計算トリガ(混雑度に対するトリガ)	0.1
RecalcThresholdForUtility1	実数	経路再計算トリガ(Utility1 に対するトリガ)	0
RecalcThresholdForUtility2	実数	経路再計算トリガ(Utility2 に対するトリガ)	0
MinRouteRecalcInterval	実数	経路再計算トリガの最小間隔	0
RouteRecalcProbability	実数	経路再計算トリガを満たした場合の経路検索確率	0
RecalcProbWhenMissingAVehicle	実数	公共交通車両(電車、バス)に乗りなかった場合の経路検索確率	0
FuelConsumption	実数	ガソリンコスト	0
BoardingDelay	実数	バスの乗り込み時間[s]	0
PedestrianRecognitionProbability	実数	ドライバーが歩行者を認識する確率	1
TimeHeadway	実数	ドライバーの前方車両との時間間隔(ヘッドウェイ)[s]	1.5
MinVehicleGap	実数	ドライバーの最小車間距離[m]	3.0
MaxAcceleration	実数	ドライバーの最大加速度[m/s^2]	1.0
MaxDeceleration	実数	ドライバーの最大減速度[m/s^2] (※減速度は負の値で指定する)	-3.0
MaxBrakingDeceleration	実数	車両の最大減速度[m/s^2]	-20
AccelerationExponent	実数	ドライバーの加速度指数	4
SaveDeceleration	実数	車線変更による後方車両の減速度の閾値[m/s^2]	-12

MinTurnSpeed	実数	カーブでの最小速度[m/s]	10
LaneChangeAccelerationThreshold	実数	ドライバの車線変更加速度閾値[m/s ²]	0.2
VelocityRatioGapDistance	実数	IDM モデル[1]における係数 s_1 (通常の IDM モデルでは s_1 の項は 0)	0
OtherVehicleEntranceTime	実数	交差点において、相手車両が交差点に進入しようとしていることを判断する時間距離[s]	1
PassiveYieldTime	実数	交差点での譲り合いの際に他車両に対して消極的に譲歩する場合の速度差の閾値、最大の時間[s]	3.0
ActiveYieldTime	実数	交差点での譲り合いの際に他車両に対して積極的に譲歩する場合の速度差の閾値、最大の時間[s]	-3.0
YieldWaitingTime	実数	譲歩時間[s]	1
TravelDistance	実数	総移動距離[m]	0
TravelTime	実数	総移動時間[s]	0
RoadWidthWeight	実数	経路計算時の道路幅の重み	0
Disaster	実数	災害モード 災害モードが 0 の場合、歩行者は沿道移動し、1 の場合車両の有無に関係無く路上の任意の領域を移動する。	0
RouteRecalcInterval	実数	経路再計算間隔 0 以上を指定した場合、指定間隔毎に経路の再計算を行う。	0
WalkableDistanceToPrivateCar	実数	許容可能な自家用車までの歩行距離[m]	1000
WalkableDistanceToBusStopOrStation	実数	許容可能な駅・バス停までの歩行距離[m]	1000
MinDistanceToUseVehicle	実数	車・タクシーを利用する場合の	0

		最小の距離[m]	
MaxRouteCandidates	整数	料金、移動時間、乗り換え回数の各メトリックの経路について、検索する経路候補の最大数	3
NumberOfPeople	整数	人数[人]	1
MaxWaitingTimeAtEntrance	実数	入り口での最大待ち時間[s]	0
MaxWaitingTimeAtVehicleEntrance	実数	自家用車ドライバの入り口での最大待ち時間[s]	0
PrivateCarOwnership	実数	車の所有率	0
BicycleOwnership	実数	自転車の所有率	0
WalkSpeedAtTransfer	Normal Slow VerySlow WheelChair	公共交通機関の乗り換えの移動に要する時間 Normal: 3[m/s] Slow: 1[m/s] VerySlow 、 WheelChair : 0.5[m/s]	Normal
MaxWaitingTimeForTaxiAssignment	実数	タクシーの割り当てを待つ最大の時間	600
任意のパラメータ (※計算式で利用可能)	実数		0

ルート検索の設定

ルート検索は以下のパラメータに式を与えて設定します。

RoutePriority

ルート検索の計算式一覧

式名	説明
_MoveByPreferedMobilityMeans	優先移動手段
_ArrivalTime	到着時間 [s]
_TotalTravelTime	合計移動時間 [s]
_TotalTravelDistance	合計移動距離 [m]
_TotalPublicTransportationDelay	合計遅延時間 [s]
_Price	合計運賃

_TotalTransferCount	合計乗り換え回数
_TotalTransferDuration	合計乗り換え時間 [s]
_NumberOfPeopleOnRoute	経路検索時点での経路上の歩行者数の総和
_NumberOfVehiclesOnRoute	経路検索時点での経路上のレーンあたりの車両数の総和

その他の計算式一覧

式名	説明
LOG10(value)	常用対数
LOG(value)	自然対数
POW(value, multiplier)	べき乗
MIN(value1, value2)	最小
MAX(value1, value2)	最大
SQRT(value)	平方根
SIN(value)	サイン(正弦)
COS(value)	コサイン(余弦)
TAN(value)	タンジェント(正接)
ABS(value)	絶対値
CEIL(value)	切り上げ
FLOOR(value)	切り捨て
PI()	パイ(円周率)
EXP(value)	指数
UNI(min, max)	整数一様分布
UNID(min, max)	倍精度浮動小数点一様分布
NORMAL(average, deviation)	正規分布
EXPDIST(lambda)	指数分布
POISSON(lambda)	ポアソン分布
ERLANG(lambda, phase)	アーラン分布

3.3. エージェント行動設定ファイル

エージェントの行動を設定します。

「BehaviorType」タグで行動タイプを設定します。

BehaviorType : <行動タイプ名>

ある「BehaviorType」タグから次の「BehaviorType」タグ、またはファイル終了までのパラメータ設定を当該プロファイルタイプの行動設定とします。

ある行動を以下のように記述します。

[時間設定] <イベント>

例) シミュレーション開始時刻に building1 から building2 へ移動を開始する。

[-] InitialLocation = building1, MoveToDestination = building2

イベント一覧

イベント名	説明
[Time]	ある行動の開始時刻を設定します。各行動設定の先頭に[]で記述します。HH:MM、またはシミュレーション開始からの時間(秒)で指定。[-] は前の行動が終了次第開始することを意味し、省略した場合と同じ。
InitialLocation	指定する初期位置/ロケーショングループを基点に Agent を出現させる
InitialLocationId	指定する初期位置 ID に Agent を出現させる
MoveToDestination	指定する目的地/ロケーショングループ内の目的地に移動する
MoveToDestinationId	指定する目的地 ID に移動する
GenerateApplication	通信アプリケーションの設定

オプションパラメーター一覧

パラメータ名	説明
DepartureTime	出発時間
ArrivalTime	到着時間
Wait	待機時間
WaitUntil	最小行動継続時刻(行動が終了する最小時間)
LocationGroup	ロケーショングループ
LocationIdGroup	ロケーション ID グループ
InterruptCurrentAction	割り込み指定 [Time]で指定した時刻の割り込み行動を行う。 ResumeAfterInterruption: 割り込み行動を終了後、割り込み前の行動に戻る TerminateNow: 割り込み行動を終了後、割り込み前の行動に戻らず、次に予定していた行動を実施する。
DestinationChoiceType	MoveToDestination にロケーショングループが設定されている場合の目的地選択方法(Random/Nearest) Random: 目的地の集合からランダムで目的地を選択 Nearest: 移動開始時点で最も近い目的地を選択

DestinationBaseLocation	DestinationChoiceType で Nearest を設定した際の、基準となる場所
MobilityMeans	移動手段の指定
PreferedMobilityMeans	優先的に利用する移動手段の指定
IntersectionToGoThrough	通過する交差点の指定

予約語一覧

予約語	説明
RandomBuilding	無作為の建物
RandomPark	無作為の公園
RandomPoi	無作為の POI
Home	最初に指定された Location が Home となる
PresentLocation	InitialLocation のイベントで指定することで、シナリオ初期位置エージェントが配置される
AgentLocation	DestinationBaseLocation のオプションに指定留守ことで、エージェントの位置が基準位置となる

3.4. エージェントタイムテーブル設定ファイル

バスや電車などの交通機関のタイムテーブルを設定します。

タイムテーブルは交通機関の種別、車両の形状などを記述する「Family」句と、路線名、運賃、時刻表などを記述する「Line」句により構成されます。

「Family」句のフォーマット

"Family", <vehicle family name>, <width>, <length>, <capacity>

「Line」句のフォーマット

"Line", <line name>, <"Train" or "Bus">

"LoopLine", <true/false>, <interval>, <end time>

"Price", <Price/Distance(meters)>

"Stop", <stop station name1>, <stop station name2>,...

"Route", <pass intersection name1>:<pass intersection name2>..., <stop station name3>,...

for "Bus" <vehicle family name>, <arrival time>-"<departure time>,...

4. プロパティ

4.1. プロパティ一覧

以下はコンフィギュレーションファイルで定義可能な Scenargie Multi-Agent Extension Module 固有プロパティです。

コンフィギュレーションファイルの記述方法および Scenargie Base Simulator 共通プロパティについては Scenargie Base Simulator ユーザガイド参照のこと。

型が時間の場合、時間＋単位：S(秒)、MS(ミリ秒)、US(マイクロ秒)、NS(ナノ秒)となります。

尚、デフォルト値は、コンフィギュレーションファイルにパラメータが記載されていない場合に使用される初期値であり、VisualLab で初期設定される値とは異なります。「なし」と記述されているパラメータはモデル内での初期値が存在せず、何らかの値の設定が必要な項目になります。

MultiAgent 設定

パラメータ名	スコープ	型	デフォルト値	説明(範囲:単位)
multiagent-profile-file	グローバル	文字列	なし	エージェントプロファイル定義ファイルの指定
multiagent-behavior-file	グローバル	文字列	なし	エージェント行動定義ファイルの指定
gis-public-vehicle-file	グローバル	文字列	なし	エージェント時刻表ファイルの指定
multiagent-profile-value-output-file	グローバル	文字列	なし	シミュレーションで実際に使用されたエージェントプロファイルを出力する出力ファイルの指定
multiagent-start-time	グローバル	時間	なし	時刻表へのオフセット時間
multiagent-navigation-system-update-interval	グローバル	時間	なし	経路検索に利用する統計情報(経路上のエージェント数等)の更新間隔
multiagent-profile-type	ノード	文字列	なし	エージェントのプロファイルタイプ指定
multiagent-behavior-type	ノード	文字列	なし	エージェントの行動タイプ指定

GIS 設定 (.config)

パラメータ名	スコープ	型	デフォルト値	説明(範囲:単位)
gis-los-break-down-curved-road-into-straight-roads	グローバル	Bool	false	道路の分割指定
gis-number-entrances-to-building	グローバル	整数	0	建物への最小の入り口数
gis-number-entrances-to-station	グローバル	整数	0	駅への最小の入り口数
gis-number-entrances-to-busstop	グローバル	整数	0	バス停への最小の入り口数
gis-number-entrances-to-park	グローバル	整数	0	公園への最小の入り口数
gis-road-set-intersection-margin	グローバル	Bool	false	交差点領域の作成 ※マルチエージェントシミュレーションでは必ず true に設定
gis-trafficlight-pattern-definition-file	グローバル	文字列	なし	信号パターン定義ファイル
gis-object-enable-time	ノード (GIS オブジェクト)	整数	なし	GIS オブジェクトが有効となる時刻
gis-object-disable-time	ノード (GIS オブジェクト)	整数	なし	GIS オブジェクトが無効となる時刻
gis-entrance-queue-max-flow-rate-people-per-sec	ノード (GIS オブジェクト)	実数	無限	入口での一秒当たりの最大流入可能人数 [人/s]
gis-entrance-queue-type	ノード (GIS オブジェクト)	文字列	Line	入口待ちのキュー種別
gis-entrance-line-per-row	ノード (GIS オブジェクト)	整数	3	入口待機での列あたりの人数 [人/列]
gis-entrance-line-per-separation	ノード (GIS	実数	1.0	入口待機列の間隔

	オブジェクト)			[m]
gis-entrance-line-column-separation-meters	ノード (GIS オブジェクト)	実数	1.0	入口待機行の間隔 [m]

GIS 設定 (.shp)

パラメータ名	型	デフォルト値	説明(範囲:単位)
id	整数	なし	オブジェクト ID
name	文字列	なし	名前
width	実数	1	道路幅
height	実数	なし	建物高さ
type	文字列	なし	道路種別
Lane12	整数	1	始点から終点向きの車線数
Lane21	整数	1	終点から始点向きの車線数
capacity	実数	なし	道路: 歩行者のキャパシティ 建物: 人の最大収容人数
vehiclecap	実数	なし	車の最大収容数
speedlimit	実数	なし	制限速度
signaloff	実数	なし	開始時刻オフセット
green	実数	なし	青色時間
yellow	実数	なし	黄色時間
red	実数	なし	赤色時間
pattern	文字列	なし	制制御パタン名
objected	ノード(GIS オブジェクト ID)	なし	入り口の関連付けられた道路のオブジェクト OD
roadid	ノード(GIS オブジェクト ID)	なし	信号の関連付けられた道路のオブジェクト OD

5. 統計値、トレース

5.1. 統計値の標準設定一覧表

レイヤ	モデル名	統計値	説明
MAS	Mas	Population	滞在人数
		Utility1	効用 1
		Utility2	効用 2
		Congestion	混雑度(エージェント数/ オブジェクトのポリゴン面 積 [m ²]、バス 停 の み 1[m ²])
		TravelDistance	総移動距離 [m]
		TravelTime	総移動時間 [秒]
		NoRoutes	経路が見つからなかった 回数
		RouteCalculationTimes	経路計算回数

5.2. トレースの標準設定一覧表

レイヤ	モデル名	トレースイベント	追加情報	イベント説明
MAS	Mas タグ名: Mas	Utility1	効用 1	効用 1 の変化
		Utility2	効用 2	効用 2 の変化
		TravelDistance	移動距離 [m]	総移動距離の変 化
		TravelTime	移動時間 [秒]	総移動時間の変 化
		Destination	目的地	目的地の変化
		DestinationChange Count	目的地変更回数	目的地変更回数 の変化
		DestinationChange CountByCommunica tion	通信による目的 地変更回数	通信による目的 地変更回数の変 化
		Gis_Congesion	混雑度 [1m ² あた りの Human の数]	混雑度の変化
	タグ名: Gis	State	オブジェクトが有効 か無効か	オブジェクトの有 効/無効の変化

6. 参考文献

1. M. Treiber, A. Hennecke and D. Helbing, "Congested traffic states in empirical observations and microscopic simulations," *Physical Review E* 62, no. 2, pp. 1805-1824, 2000.

