

別紙：自動運転制御モデル
IF ガイドライン準拠
ジェネリックモデルシミュレータ
環境構築方法及び操作手順書
(Ver2.0)

改訂履歴

Rev.	日付	内容	会社名
1.0	2022/03	初版	(株)両毛システムズ
2.0	2023/02	<p>環境構築方法の一部変更に伴い改訂</p> <p>第 3 章</p> <ul style="list-style-type: none"> ・ 参照先 URL を変更 ・ 図を追加 <p>第 4 章 1</p> <ul style="list-style-type: none"> ・ 「Python3.7.9 のインストール」の説明を補足 <p>第 4 章 2</p> <ul style="list-style-type: none"> ・ 「CARLA バイナリパッケージのインストール」の図を変更 <p>第 4 章 3</p> <ul style="list-style-type: none"> ・ 「MATLAB ToolBox のインストール」に図を追加 <p>第 4 章 4.2</p> <ul style="list-style-type: none"> ・ CMake のインストール作業を追記 ・ 「7-Zip 19.0 (64bit)のインストール」の URL および図を変更 ・ 「UnrealEngine4.26 のインストール」の説明を補足 <p>第 4 章 4.3</p> <ul style="list-style-type: none"> ・ 「バッチファイルの内容を一部変更」を追記 ・ 「PythonAPI のビルド」 1) の URL を変更 	(株)両毛システムズ

		<ul style="list-style-type: none"> ・ 「PythonAPI のビルド」 2) の説明を補足 ・ 「UnrealEngine へ RoadRunner インポートプラグインのインストール」の図、および説明を変更 <p>第 4 章 4.4</p> <ul style="list-style-type: none"> ・ 「マップのインポート」に図を追加 <p>第 5 章 1.2</p> <ul style="list-style-type: none"> ・ 「init_setting.m の実行」の説明を補足 <p>第 7 章</p> <ul style="list-style-type: none"> ・ 「複数 PYTHON インストール時の手順」、 「PYTHON3.7 選定理由」を追記 	

目次

1	はじめに	6
1.1	留意事項	6
1.2	著作権	6
2	動作・使用環境	7
2.1	動作環境	7
2.2	使用環境	8
3	シミュレーションで可能なこと	9
4	シミュレーション環境構築手順	9
4.1	デフォルトマップ／シナリオのみ利用、仮想環境なし	10
4.2	デフォルトマップ／シナリオのみ利用、仮想環境あり.....	12
4.3	シナリオを修正して利用、仮想環境あり	14
4.4	マップを修正して利用、仮想環境あり.....	15
4.4.1	EpicGamesとGit アカウントの連携	15
4.4.2	RoadRunner2021b, Microsoft VisualStudio のインストール	16
4.4.3	CARLA0.9.13 のインストール.....	23
4.4.4	自動運転制御モデル用マップのインポート.....	27
5	シミュレーションの実行手順.....	34
5.1	基本操作	34
5.1.1	CARLA を起動する.....	35
5.1.2	シミュレーションを実行する.....	36
5.1.3	実行状況を確認する	40
5.1.4	結果を確認する	44
5.2	シナリオとパラメータ	48
5.2.1	CRANTS ・ JARI 城里テストコース ・ 市街地テストコース.....	48
5.2.2	安全性評価フレームワーク.....	53

6	FAQ(トラブルシューティング)	66
7	APPENDIX	69
7.1	参考文献	69
7.2	複数 PYTHON インストール時の手順	69
7.3	PYTHON3.7 選定理由.....	71

1 はじめに

本書は、自動運転制御モデル I/F ガイドライン準拠ジェネリックモデル（以下、自動運転制御モデル）の環境設定および、シミュレーション実行の手順書です。環境設定にあたり、同種のソフトウェアおよびライブラリを既に使用している場合は、バージョン競合等により不具合が発生する場合があります。なお、本手順により発生したいかなる損害について、責任は負いかねますのでご了承ください。

1.1 留意事項

本自動運転制御モデルは、MATLAB/Simulink 使用経験のある方を対象としています。MATLAB/Simulink の環境設定や操作方法については省略していますのでご注意ください。

1.2 著作権

本文書の著作権は、著作者に帰属します。

著作者は、本文書の内容に関し、いかなる保証もするものではありません。万一本文書を利用して不具合等があった場合でも、著作者は一切責任を負いかねます。また、本文書に記載されている事項は予告なしに変更または廃止されることがありますので、あらかじめご了承ください。

2 動作・使用環境

2.1 動作環境

以下の環境での動作確認をしています。

OS	Microsoft Windows10 Pro 64bit 21H1
CPU	Intel Corei9-10900K
メインメモリ	32GB
ストレージ	1TB SSD
GPU	NVIDIA GeForce RTX3080Ti 12GB

OS	Microsoft Windows10 Pro 64bit 21H1
CPU	Intel Xeon E5-2620 2.10GHz
メインメモリ	32GB
ストレージ	1TB HDD
GPU	NVIDIA Quadro M2000 ビデオメモリ 4GB + 共有メモリ 16GB

OS	Microsoft Windows10 Home 64bit 21H2
CPU	Intel Corei i7-9750H 2.60GHz
メインメモリ	16GB
ストレージ	C:250GB D:1TB HDD
GPU	NVIDIA GeForce RTX2060 ビデオメモリ 6GB + 共有メモリ 8GB

※インストール時に、インターネット接続が必要です。

※メインメモリは、最低 16B 以上が必要です。

※ストレージは、使用方法により 100GB～500GB 程度が必要です。

2.2 使用環境

使用方法により、以下の4つのパターンがあります。目的に合わせて選択し環境設定を実施してください。






① デフォルトマップ/シナリオのみ利用 仮想環境なし	MathWorks MATLAB/Simulink R2021a Python3.7 / ライブラリ
② デフォルトマップ/シナリオのみ利用 仮想環境あり	MathWorks MATLAB/Simulink 2021a CARLA 0.9.13 バイナリ版+追加マップ Python3.7 / ライブラリ
③ ②でシナリオを修正して利用	MathWorks MATLAB/Simulink 2021a MathWorks AutomatedDrivingToolbox MathWorks ComputerVisionToolBox MathWorks ImageProcessingToolBox CARLA 0.9.13 バイナリ版+追加マップ Python3.7 / ライブラリ
④ ③でマップを修正して利用	MathWorks MATLAB/Simulink 2021a MathWorks AutomatedDrivingToolbox MathWorks ComputerVisionToolBox MathWorks ImageProcessingToolBox Mathworks RoadRunner 2021b CARLA 0.9.13 ソース+追加マップ Python3.7 / ライブラリ Microsoft VisualStudio 2019 EpicGames UnrealEngine4.26 ソース EpicGames アカウント Git アカウント

3 シミュレーションで可能なこと

本モデルでは、「自動車開発におけるプラントモデル I/F ガイドライン」※に準拠したプラントモデルに、簡易自動運転コントローラを搭載した自動運転制御モデルを接続して、あらかじめ登録されている仮想環境内での走行シナリオを実行することができます。また、自動運転評価フレームワークに基づいたいくつかの評価シナリオを実行することができます。

RoadRunner を利用したマップのカスタマイズや、AutomatedDrivingToolBox の DrivingScenarioDesigner を利用した走行シナリオのカスタマイズが可能ですが、本書の範疇から外れるため、詳細な解説は省略します。

※ 参照：<https://www.jambe.jp/system/download> の No.20002

No	カテゴリ	対象1	対象2	種類	名称	作成機関	最終作成年月	ダウンロード
20001	モデル接続/Fガイドライン	車両	車両全体	ガイドラインおよび準拠モデル	平成28年度の取組（経済産業省プレスリリースにて公開）（ https://warp.da.ndl.go.jp/info:ndljp/pid/10341576/www.meti.go.jp/press/2016/03/20170331010/20170331010.html ）	経産省MBD研究会	2018/03	
20002	モデル接続/Fガイドライン	車両	車両全体	ガイドライン	自動車開発におけるプラントモデル/Fガイドライン※英語版あり	経産省MBD研究会 ガイドライン構築委員会	2021/03	
20003	モデル接続/Fガイドライン	システム	エンジン	ガイドライン	・エンジンシステムにおけるプラントモデル/Fガイドライン（PDF形式）	AICE	2021/03	
20004	モデル接続/Fガイドライン	システム	トランスミッション	ガイドライン	動力伝達システムにおけるプラントモデル/Fガイドライン	TRAMI	2021/03	
20005	モデル接続/Fガイドライン	システム	電動化	ガイドライン	EV・電動化車両の電マネシステムにおけるプラントモデル/Fガイドライン	デジタルツインズ	2021/03	

4 シミュレーション環境構築手順

以下にシミュレーション環境構築手順を示します。前述の使用方法に合わせた手順を実施してください。

なお、本書は全て C ドライブにインストールを行うことを想定して記載しておりますが、インストール先のドライブやフォルダは任意に変更することができます。変更部分を適宜読み替えて実施してください。また、Python を複数バージョン使用している場合は、MATLAB から下記手順でインストールする Python を利用できるように pyenv 等のコマンド及び PC の環境設定を実施してください。

インストール途中でファイアウォールの変更画面が表示された場合は、許可の設定を行ってください。また、Windows のランタイムが不足している場合は、メッセージが表示されますので、Microsoft 社よりダウンロードし、インストールを実施してください。

4.1 デフォルトマップ／シナリオのみ利用、仮想環境なし

以下の手順でインストールを行います。途中ファイアウォールの変更画面が表示された場合は、許可の設定を行ってください。また、Windows のランタイムが不足している場合は、メッセージが表示されますので、Microsoft 社よりダウンロードし、インストールを実施してください。

① 準備

MATLAB/Simulink のインストーラ及びライセンス
自動運転制御モデル 一式
インターネット接続環境

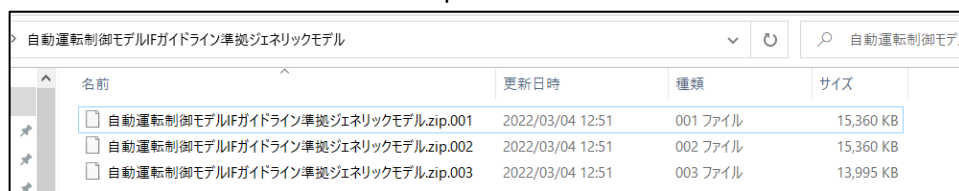
② MATLAB/Simulink のインストール

製品マニュアルを参考にし、インストールを行います。ライセンスをお持ちでない場合は、MathWorks 社様へ評価版のライセンスについてお問い合わせください。

③ モデルの展開

自動運転制御モデルをダウンロードし、解凍して任意のフォルダへ保存してください。

※ファイルが分割されている場合は、7Zip などの分割ファイルに対応したプログラムを利用して解凍してください。



名前	更新日時	種類	サイズ
自動運転制御モデルIFガイドライン準拠ジェネリックモデル.zip.001	2022/03/04 12:51	001 ファイル	15,360 KB
自動運転制御モデルIFガイドライン準拠ジェネリックモデル.zip.002	2022/03/04 12:51	002 ファイル	15,360 KB
自動運転制御モデルIFガイドライン準拠ジェネリックモデル.zip.003	2022/03/04 12:51	003 ファイル	13,995 KB

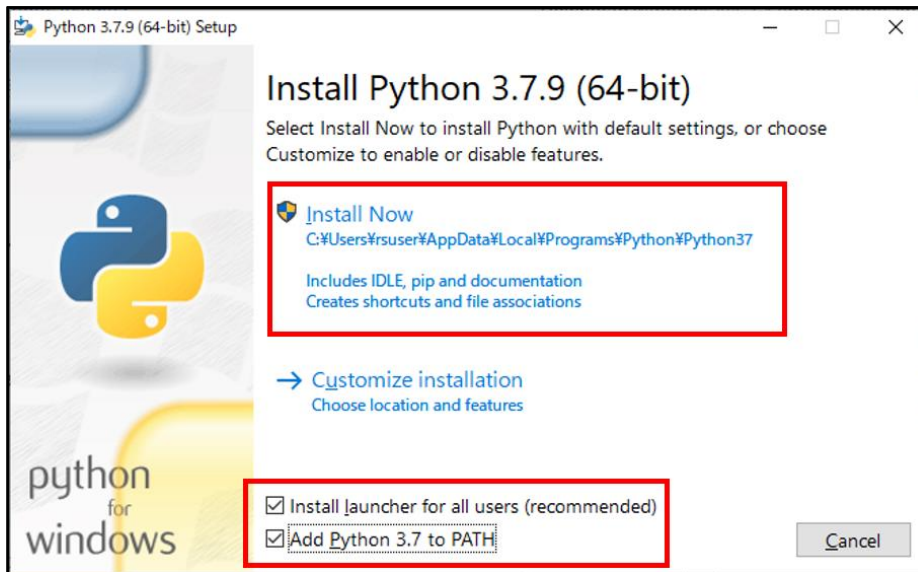
④ Python3.7.9 のインストール

<https://www.python.org/downloads/windows>

を開き、Python 3.7.9 の Windows x86-64 executable installer をダウンロードしてください。

インストーラを実行し、以下のオプションを設定して続行してください。

- Install launcher for all users にチェック
- Add Python 3.7 to PATH にチェック

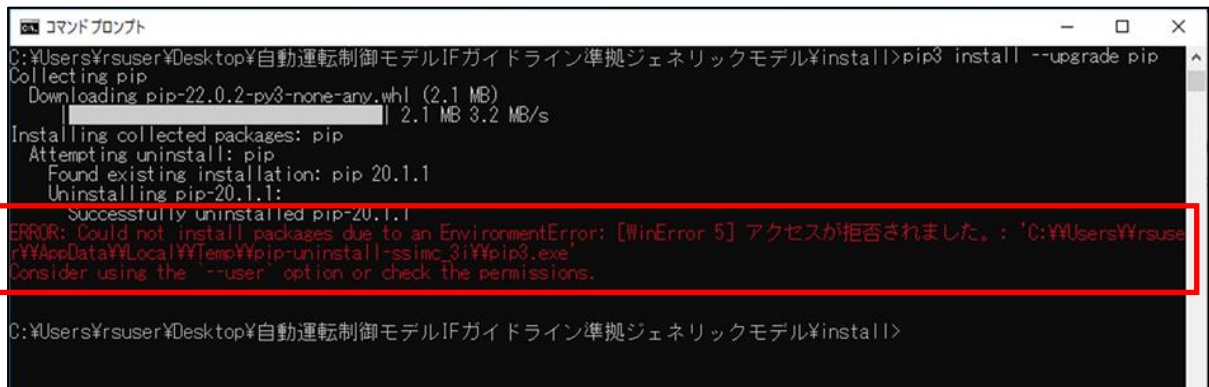


※別バージョンの Python が既にインストールされている場合、Python のバージョン変更を行ってください
(本解説書 Python のバージョンを変更する を参照ください)。

⑤ Python の環境設定

コマンドプロンプトを開き、以下のコマンドを実行してください。

```
>cd <ダウンロードした“自動運転制御モデル IF ガイドライン準拠ジェネリックモデルシミュレータ環境”フォルダ>%install  
>pip3 install --upgrade pip
```



※「アクセスが拒否されました」エラーが出た場合は、そのまま次の手順に進んでください。

```
>pip3 install --user -r requirements.txt
```

※プロキシ環境下にある場合は、pip3 コマンドの最後に、以下を追加してください。

(ただし、環境により実行できない場合があります。

その場合は、直接インターネットに接続できる環境でセットアップしてください。)

```
--proxy=http://username:password@your.proxy.address:port
```

(username:プロキシアカウント

password:パスワード

your.proxy.address:プロキシの URL

port:プロキシのポート番号)

```
コマンドプロンプト
C:\Users\¥自動運転制御モデルIFガイドライン準拠ジェネリックモデルシミュレータ環境¥install>pip3 install --user -r requirements.txt
Collecting networkx
  Downloading networkx-2.6.3-py3-none-any.whl (1.9 MB)
-----1.9/1.9 MB 3.0 MB/s eta 0:00:00
Collecting numpy
  Downloading numpy-1.21.5-cp37-cp37m-win_amd64.whl (14.0 MB)
-----14.0/14.0 MB 5.0 MB/s eta 0:00:00
```

※ワーニングが出た場合は、無視してください。

⑥ シミュレーションの実行

5 シミュレーションの実行手順へ進み、仮想環境を利用しない場合の方法でシミュレーションを実行してください。

4.2 デフォルトマップ／シナリオのみ利用、仮想環境あり

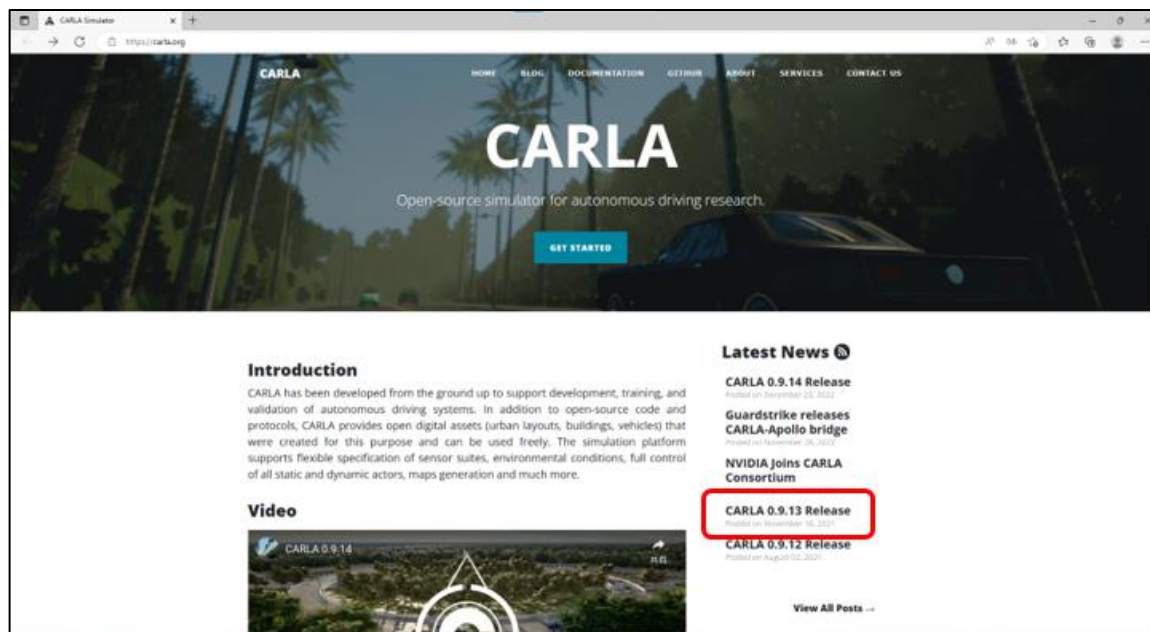
① MATLAB/Simulink および Python のインストール

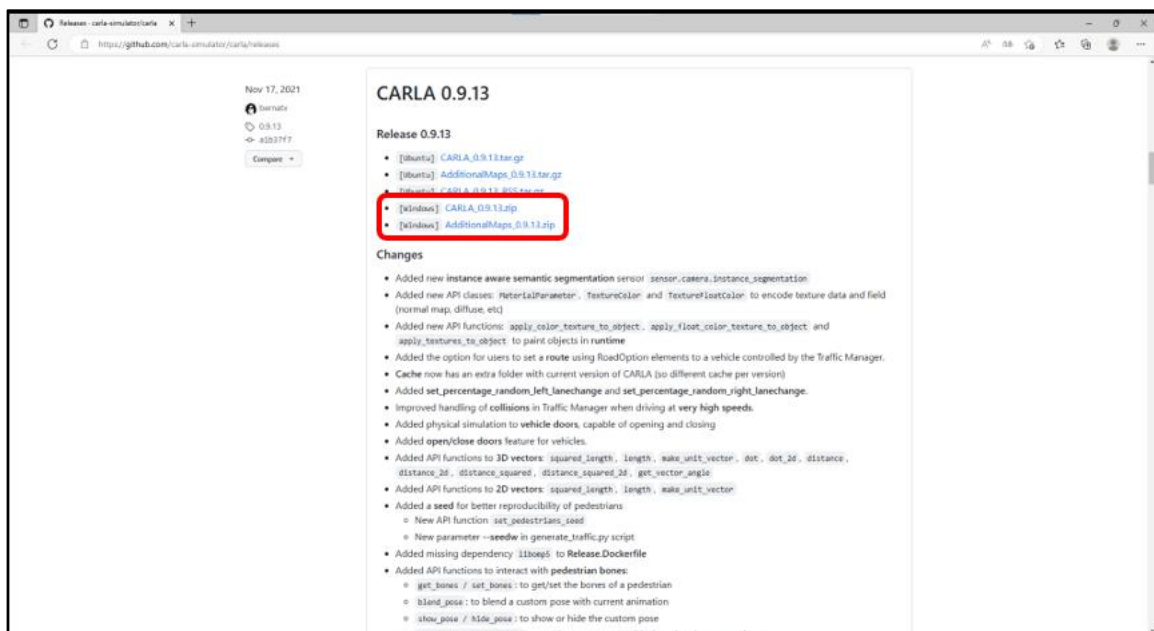
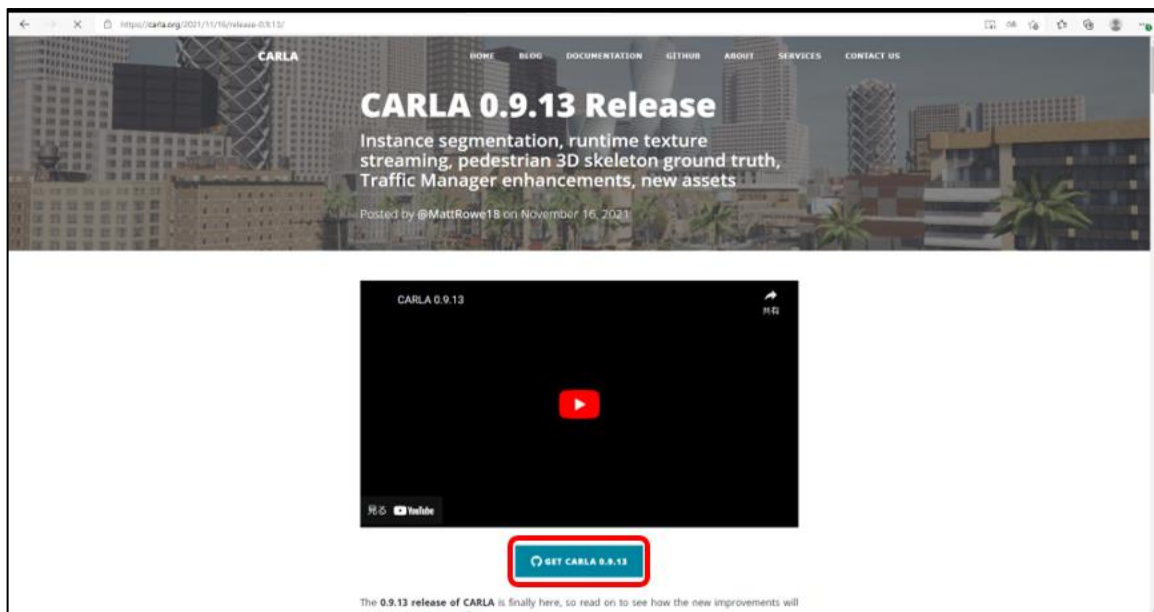
4.1①②③④⑤の方法で、MATLAB/Simulink および Python のインストールを行ってください。

② CARLA バイナリパッケージのインストール

<https://carla.org>

を開き、LatestNews または ViewAllPosts より[CARLA0.9.13 Release]をクリックし、表示されたページの [GET CARLA 0.9.13]をクリックし、[Windows]CARLA_0.9.13.zip 及び [Windows]AdditionalMaps_0.9.13.zip をダウンロードしてください。





ダウンロードした[CARLA_0.9.13.zip]ファイルを、[c:¥carla]フォルダに解凍してください。

[c:¥carla¥WindowsNoEditor]が作成されます。

※インストール先のドライブ、ファイル名は変更できます。

その際は以下の手順にて、ドライブ名“c:”またはフォルダ名“¥carla”を読み替えてください。

同様にダウンロードした [AdditionalMaps_0.9.13.zip]の中身を、先ほど作成された [c:¥carla¥WindowsNoEditor]内に、上書きコピーしてください。

③ PythonへCARLAライブラリのインストール

コマンドプロンプトを開き、以下のコマンドを実行してください。

```
>pip3 install --user c:%carla%WindowsNoEditor%PythonAPI%carla%dist%carla-0.9.13-cp37-cp37m-win_amd64.whl
```

④ 統合 ECU モデル用マップのインストール

ダウンロードした“自動運転制御モデル IF ガイドライン準拠ジェネリックモデルシミュレータ環境”フォルダ内の、[CarlaEXE 用追加ファイル%CarlaEXE 用追加ファイル.zip]内のファイルを、[c:%carla%WindowsNoEditor]内へ上書きコピーしてください。

⑤ シミュレーションの実行

5 シミュレーションの実行手順へ進んでください。

4.3 シナリオを修正して利用、仮想環境あり

① MATLAB/Simulink/Python/CARLA バイナリのインストール

4.2①②③④の方法で、MATLAB/Simulink/Python/CARLA バイナリのインストールを行ってください。

② MATLAB ToolBox のインストール

MathWorks AutomatedDrivingToolbox/ComputerVisionToolBox/ImageProcessingToolBox をインストールしてください。AutomatedDrivingToolbox に付属の DrivingScenarioDesigner が利用できることを確認してください。



③ シナリオを作成して実行

DrivingScenarioDesigner を利用してシナリオを作成し、操作マニュアルに従ってシナリオを読み込み、シミュレーションを実行してください。

※OpenSCENARIO は全ての機能には対応していません。シナリオの構成によっては、正しく読み込めない場合があります。

4.4 マップを修正して利用、仮想環境あり

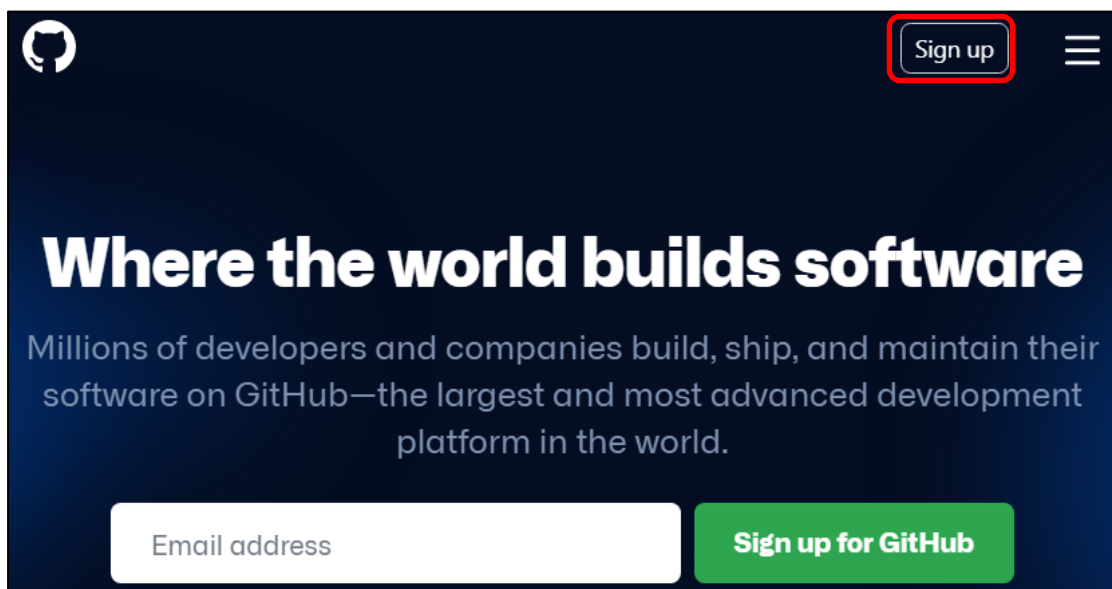
はじめに、4.2①②③及び 4.3②を実施し必要なツール等をインストールしておいてください。

4.4.1 EpicGames と Git アカウントの連携

① アカウント作成

<https://github.com/>

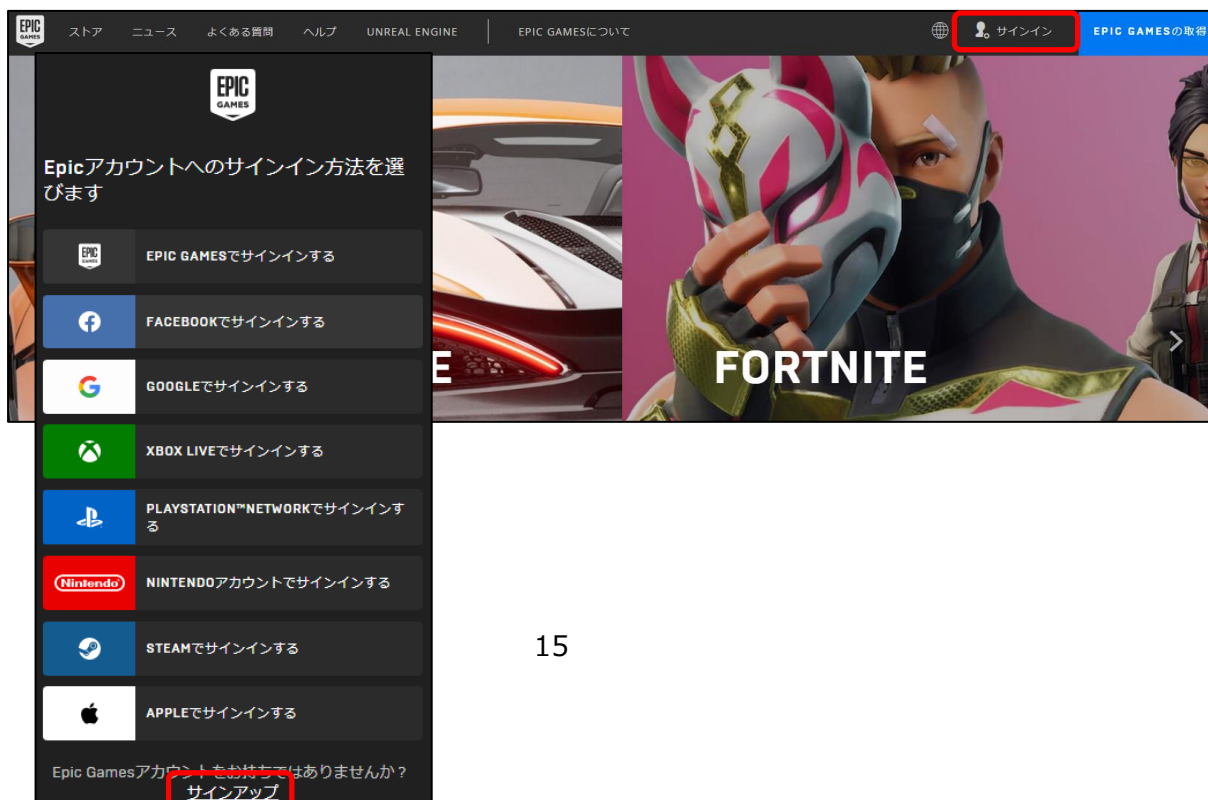
より、Git アカウントを作成してください。



② EpicGames にログイン

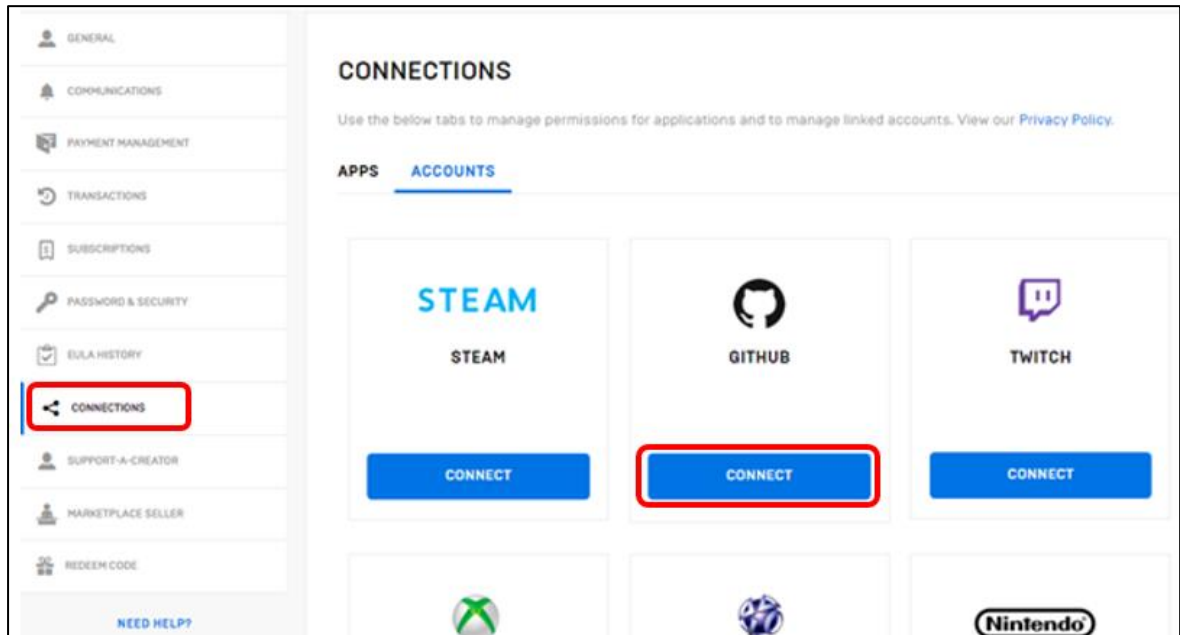
<https://www.epicgames.com/site/ja/home>

よりアカウントを作成し、ログインしてください。



右上のアカウントをクリックし、アカウントダッシュボードを開き、サイドバーから「接続済みアカウント」を開いてください。

アカウントタブの GitHub アイコンの「接続する」をクリックしてください。



クリエイターのエンドユーザーライセンスを読み、認証を続行してください。

登録したメールに、認証メッセージが届くので、[Join @EpicGames]ボタンをクリックしてください。

4.4.2 RoadRunner2021b, Microsoft VisualStudio のインストール

① RoadRunner2021b のインストール

製品マニュアルを参照し、インストールを行ってください。

② Microsoft VisualStudio 2019 Community 版のインストール

Microsoft VisualStudio 2019 Community 版をインストールします。Community 版以外のエディションでも利用可能です。

<https://visualstudio.microsoft.com/ja>

を開き、該当エディション/バージョンのプログラムをダウンロードしてください。

※2019 版は、ダウンロードタブの「以前のバージョンのダウンロード」リンクよりダウンロードできます。

ダウンロードしたファイルを実行し、以下の設定でインストールを行ってください。

- C++によるデスクトップ開発 を追加
- .NET デスクトップ開発 を追加
- .NET Framework4.6.2 開発ツール を追加
- C++によるゲーム開発 を追加



- ・ユーザー環境変数[Path]に以下のパスを追加してください。（Community版の場合）
 - ※Community版以外の場合は、下のパスを適宜読み替えてください。
 - ※Build番号により、下のパスのバージョン部分が異なる場合がありますので、適宜読み替えて下さい。
 - <インストールフォルダ>%2019%Community%VC%Tools%MSVC%14.29.30133%bin%Hostx64%x64
 - <インストールフォルダ>%2019%Community%Common7%IDE%CommonExtensions%Microsoft%CMake%CMake%bin
 - C:%Program Files (x86)%Windows Kits%10%bin%10.0.19041.0%x64
- ・ユーザー環境変数[INCLUDE]に以下のパスを追加してください。
 - C:%Program Files (x86)%Windows Kits%10%Include%10.0.19041.0%um
 - C:%Program Files (x86)%Windows Kits%10%Include%10.0.19041.0%ucrt
 - C:%Program Files (x86)%Windows Kits%10%Include%10.0.19041.0%shared
 - <インストールフォルダ>%2019%Community%VC%Tools%MSVC%14.2930133%include

- ・ユーザー環境変数[LIB]に以下のパスを追加してください。

C:\Program Files (x86)\Windows Kits\10\Lib\10.0.19041.0\um\x64

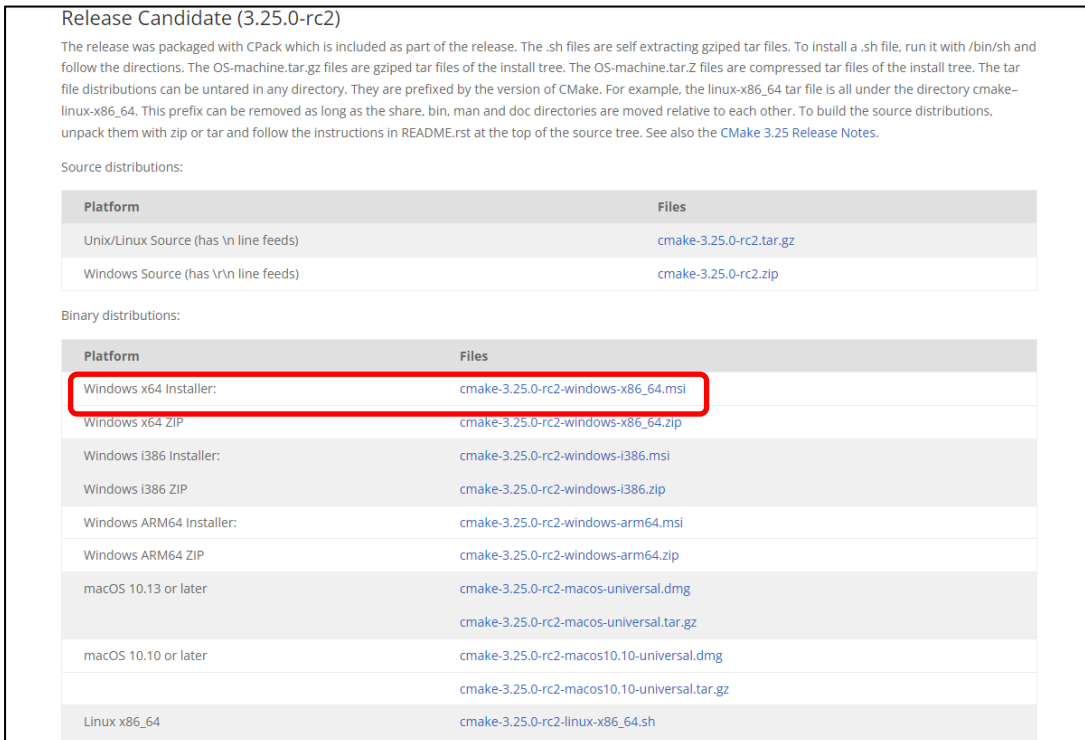
C:\Program Files (x86)\Windows Kits\10\Lib\10.0.19041.0\ucrt\x64

<インストールフォルダ>\2019\Community\VC\Tools\MSVC\14.2930133\lib\x64

③ CMake のインストール

- ・<https://cmake.org/download/>

を開き、Windows x64 Installer をダウンロードしてください。



Release Candidate (3.25.0-rc2)

The release was packaged with CPack which is included as part of the release. The .sh files are self extracting gzipped tar files. To install a .sh file, run it with /bin/sh and follow the directions. The OS-machine.tar.gz files are gzipped tar files of the install tree. The OS-machine.tar.Z files are compressed tar files of the install tree. The tar file distributions can be untared in any directory. They are prefixed by the version of CMake. For example, the linux-x86_64 tar file is all under the directory cmake-linux-x86_64. This prefix can be removed as long as the share, bin, man and doc directories are moved relative to each other. To build the source distributions, unpack them with zip or tar and follow the instructions in README.rst at the top of the source tree. See also the [CMake 3.25 Release Notes](#).

Source distributions:

Platform	Files
Unix/Linux Source (has \n line feeds)	cmake-3.25.0-rc2.tar.gz
Windows Source (has \r\n line feeds)	cmake-3.25.0-rc2.zip

Binary distributions:

Platform	Files
Windows x64 Installer:	cmake-3.25.0-rc2-windows-x86_64.msi
Windows x64 ZIP	cmake-3.25.0-rc2-windows-x86_64.zip
Windows i386 Installer:	cmake-3.25.0-rc2-windows-i386.msi
Windows i386 ZIP	cmake-3.25.0-rc2-windows-i386.zip
Windows ARM64 Installer:	cmake-3.25.0-rc2-windows-arm64.msi
Windows ARM64 ZIP	cmake-3.25.0-rc2-windows-arm64.zip
macOS 10.13 or later	cmake-3.25.0-rc2-macos-universal.dmg cmake-3.25.0-rc2-macos-universal.tar.gz
macOS 10.10 or later	cmake-3.25.0-rc2-macos10.10-universal.dmg cmake-3.25.0-rc2-macos10.10-universal.tar.gz
Linux x86_64	cmake-3.25.0-rc2-linux-x86_64.sh

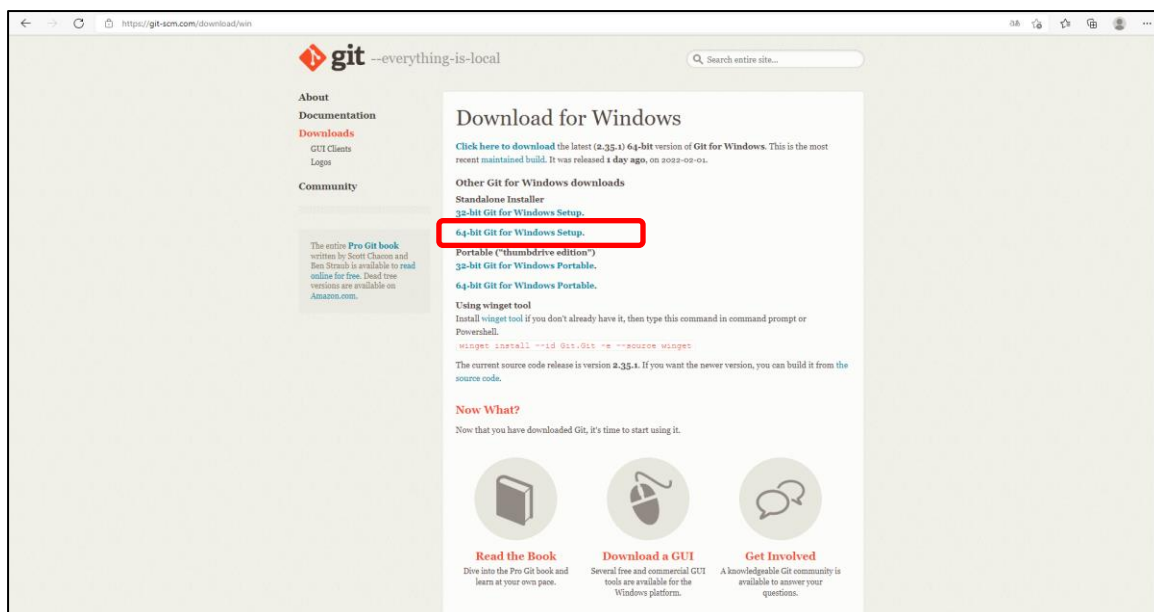
- ・ダウンロードしたファイルを実行し、インストールしてください。

※バージョン 3.24.2 までは以降の作業で動作することを確認しました。

④ Git のインストール

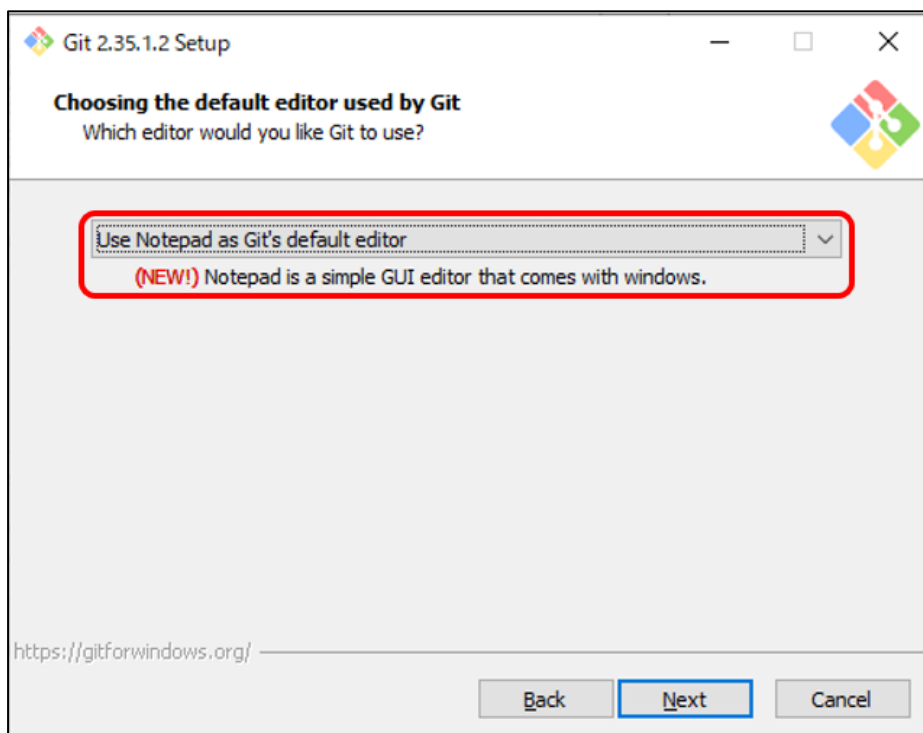
<https://git-scm.com/download/win>

を開き、「64-bit Git for Windows Setup.」をダウンロードしてください。



ダウンロードしたファイルを実行し、以下の設定でインストールしてください。

- ・DefaultEditor 設定で、notepad を選択（任意）



※上記以外のオプションはデフォルト設定でインストールしてください。

- ・ユーザー環境変数[Path]に以下のパスを追加してください。

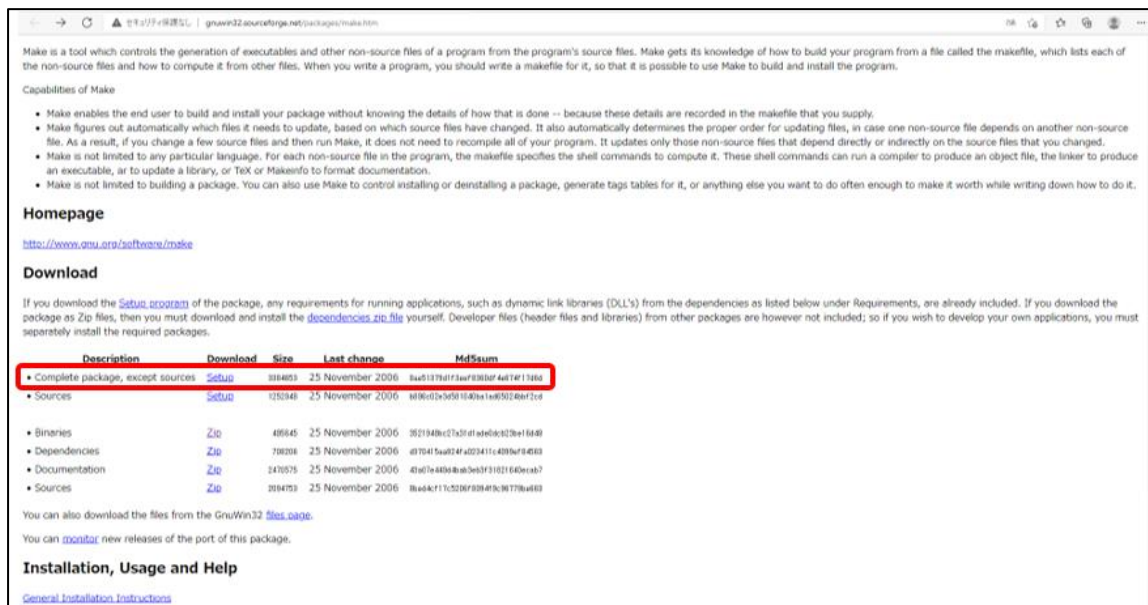
<インストールフォルダ>%bin

※バージョン 2.38.1 までは以降の作業で動作することを確認しました。

⑤ Make 3.81 のインストール

<http://gnuwin32.sourceforge.net/packages/make.htm>

を開き、該当ファイルをダウンロードしてください。



ダウンロードしたファイルを実行しインストールしてください。

※オプションはデフォルト設定でインストールしてください。

・ユーザー環境変数[Path]に以下のパスを追加してください。

<インストールフォルダ>%bin

⑥ 7ZIP 19.0 (64bit)のインストール

<https://sevenzip.osdn.jp/download.html>

を開き、該当ファイルをダウンロードしてください。



ダウンロードしたファイルを実行しインストールしてください。

※オプションはデフォルト設定でインストールしてください。

⑦ UnrealEngine4.26 のインストール

・コマンドプロンプトを開き、以下のコマンドを実行し、ファイルをダウンロードしてください。

※異なるドライブにインストールする場合は、適宜読み替えてください。

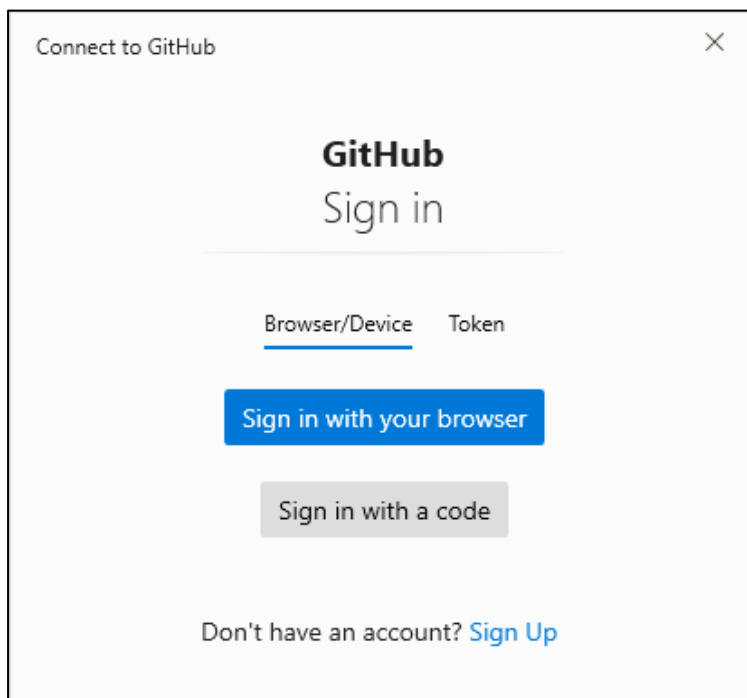
```
>mkdir c:¥carla¥UE4_26
```

```
>cd c:¥carla¥UE4_26
```

```
>git clone --depth 1 -b carla https://github.com/CarlaUnreal/UnrealEngine.git .
```

※インストールに時間がかかります。スリープしないよう注意してください。

※以下の GitHub のログイン画面が表示された場合は、前項で作成したアカウントを利用してログインしてください。



```
>Setup.bat
```

※インストールに時間がかかります。スリープしないよう注意してください。

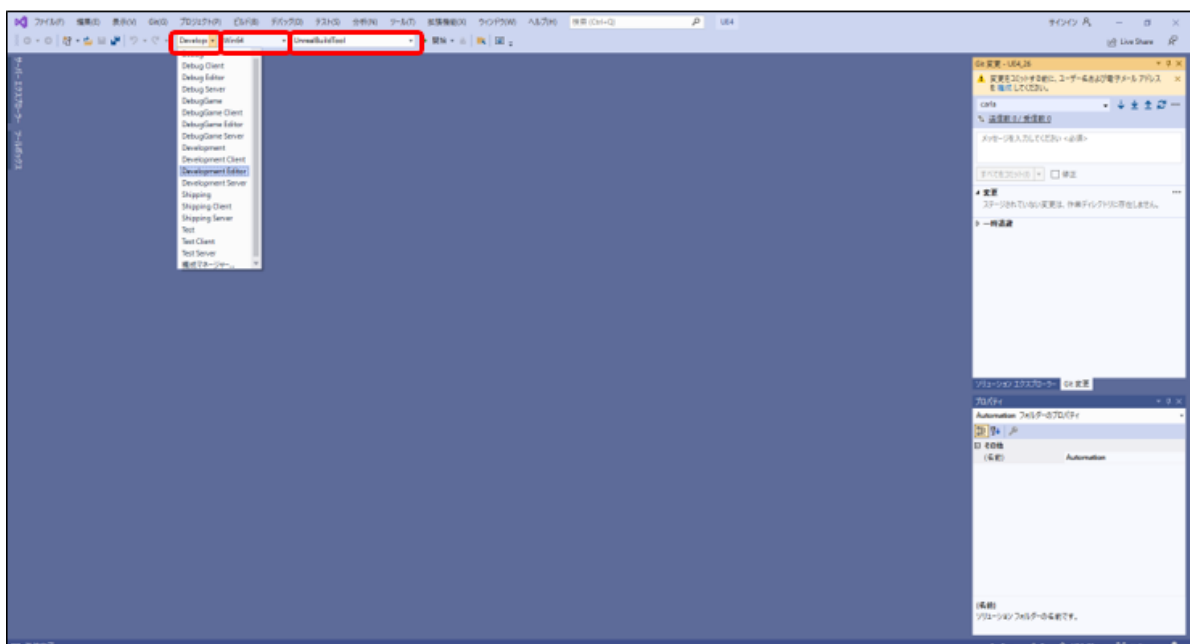
```
>GenerateProjectFiles.bat
```

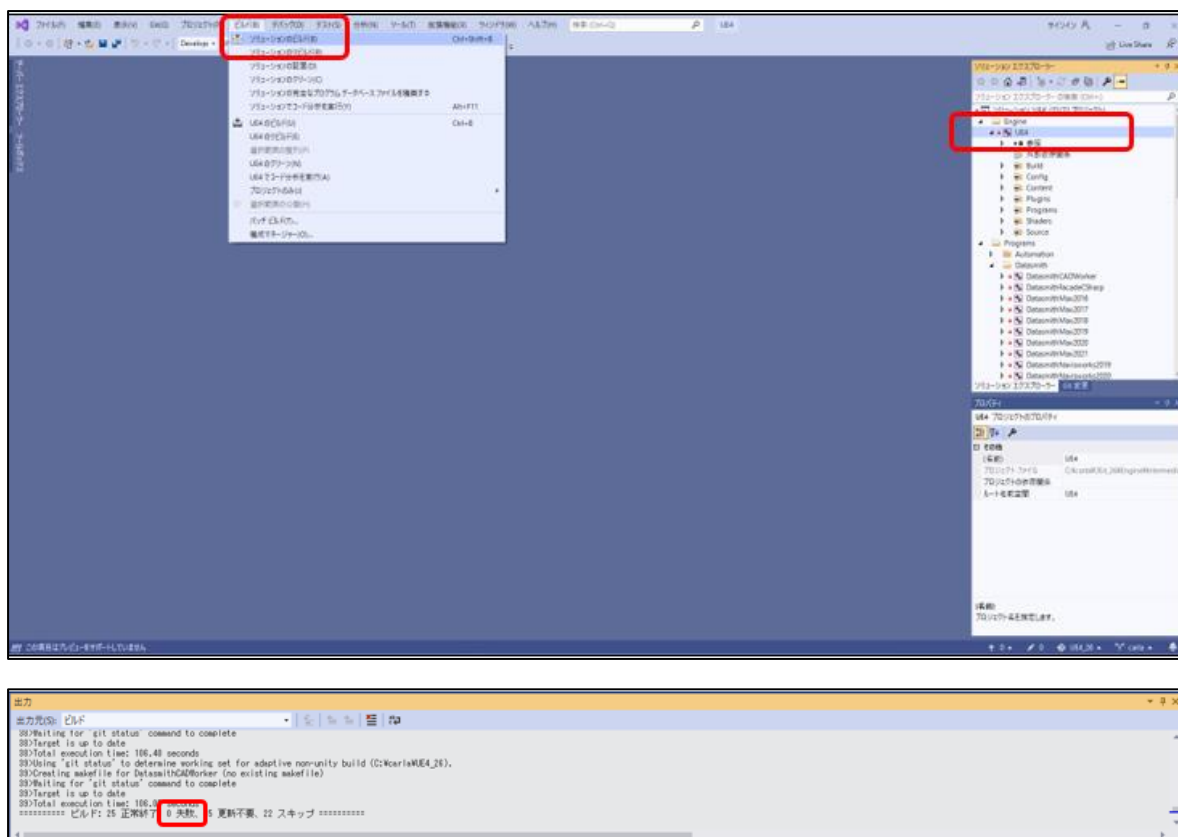
※以下の画面が表示された場合は、インストールを実施してください。



・c:\¥carla¥UE4_26¥UE4.sln をダブルクリックして、VisualStudio を起動し、以下の設定でビルドしてください。

- ・ソリューションの構成 : Development Editor
- ・ソリューションプラットフォーム : Win64
- ・スタートアッププロジェクト : UnrealBuildTool
- ・ソリューションで、UE4 を選択





・エラーが無いことを確認してください。

※エラーが生じた場合、環境変数が正しく設定できているか確認してください。

・ユーザー環境変数[UE4_ROOT]に以下のパスを追加してください。

c:%carla%UE4_26

・以下のファイルが作成されていることを確認してください。

c:%carla%UE4_26%Engine%Binaries%Win64%UE4Editor.exe

4.4.3 CARLA0.9.13 のインストール

① Carla ファイルのダウンロード

コマンドプロンプトを新しく開きなおい、以下のコマンドを実行し、ファイルをダウンロードしてください。

※ビルド中にエラーが発生する場合は、VisualStudio メニュー内にある、

「x86_x64 CrossToolCommandPrompt」のコマンドプロンプトを使用してください。

>mkdir c:%carla%carla

>cd c:%carla%carla

>git clone --depth 1 -b 0.9.13 https://github.com/carla-simulator/carla .

>Update.bat

※インストールに時間がかかります。スリープしないよう注意してください。

② バッチファイルの内容を一部変更

- c:¥carla¥carla¥Util¥InstallersWin¥install_zlib.bat を開いてください。
- install_zlib.bat 51 行目の“set ZLIB_VERSION = 1.2.11”を“set ZLIB_VERSION = 1.2.13”に変更してください。
- ※ZLIB_VERSION は下記ホームページを参考に、適宜最新のものに変更してください。
<https://www.zlib.net/>
- Python3.7 以外のバージョンがインストールされている場合、c:¥carla¥carla¥Util¥BuildTools 内の BuildPythonAPI.bat、BuildCarlaUE4.bat、Package.bat の内容を一部変更してください(本解説書 バッチファイルの修正 を参照ください)。

③ PythonAPI のビルド

以下のコマンドを実行し、PythonAPI をビルドしてください。

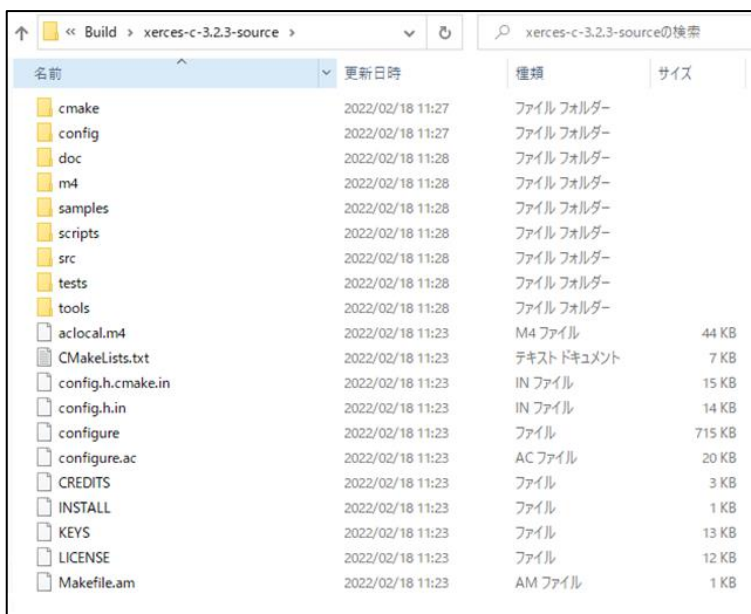
>make PythonAPI

- 以下のファイルが作成されていることを確認してください。

【c:¥carla¥carla¥PythonAPI¥carla¥dist 内に、[* .whl][* .egg]】

※上記フォルダまたはファイルが作成されていない場合は、以下を実行してください。

- 1) <https://archive.apache.org/dist/xerces/c/3/sources/>を開き、xerces-c-3.2.3.zip をダウンロードしてください。
- 2) c:¥carla¥carla¥Build¥xerces-c-3.2.3-source 及び xerces-c-3.2.3-install の中身を削除してください。
※c:¥carla¥carla¥Build¥xerces-c-3.2.3-source 及び xerces-c-3.2.3-install が存在しない場合、c:¥carla¥carla¥Build に xerces-c-3.2.3-source という名前でフォルダを新規作成してください。
- 3) 解凍したファイル内の xerces-c-3.2.3 以下のファイル全てを、c:¥carla¥carla¥Build¥xerces-c-3.2.3-source フォルダにコピーしてください。
- 4) 以下のコマンドを再度実行してください。
>make PythonAPI



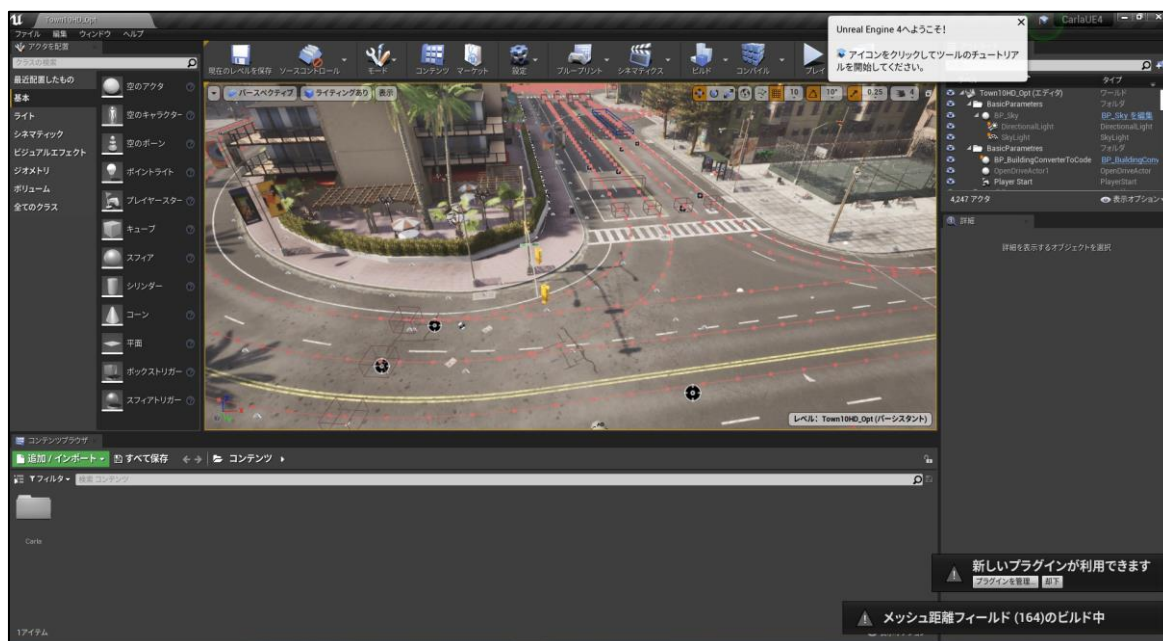
④ CARLA のビルド

以下のコマンドを実行し、CARLA をビルドしてください。

```
>make launch
```

・ビルドに成功した場合、CARLA が起動します。

※ファイアウォールの警告が出た場合は、許可してください。



※画面右下に、ビルド中のメッセージが出ている場合は、終了するまでお待ちください。

⑤ UnrealEngine の設定

UnrealEngine の[編集]-[エディタの環境設定]の左側のバーにある[パフォーマンス]を選択し、「バックグラウンド

での処理により少ない CPU を使用」のチェックを外してください。



・UnrealEngine を閉じてください。

⑥ Python へ CARLA ライブラリのインストール

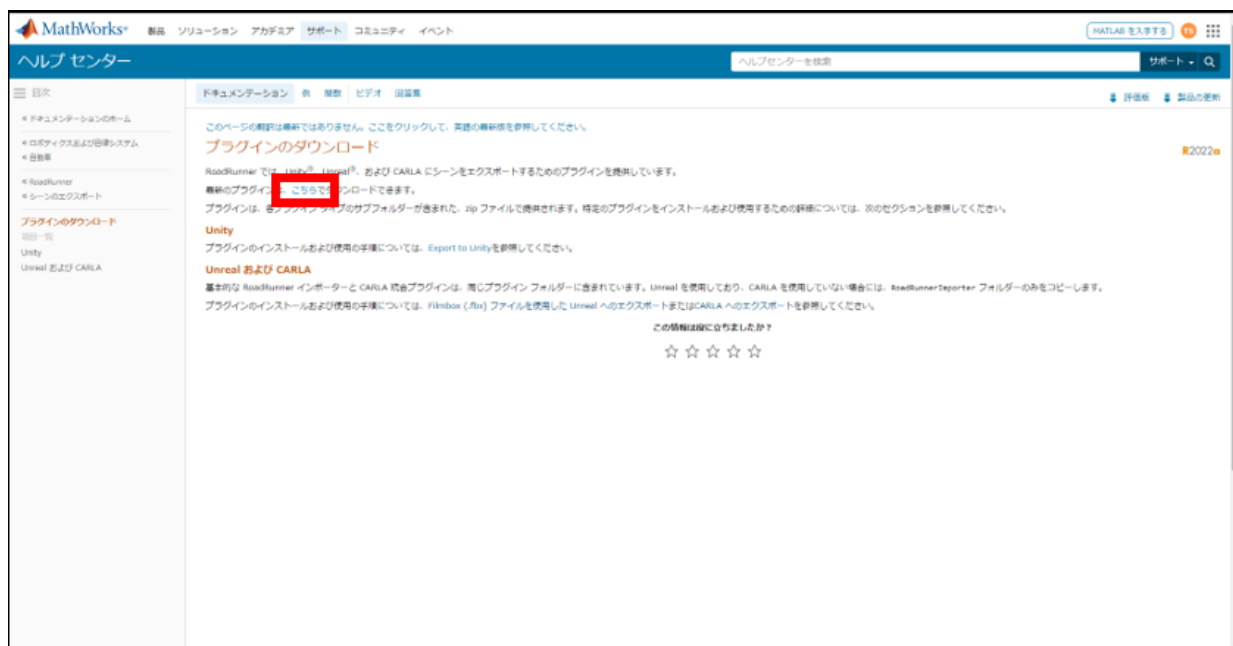
以下のコマンドを実行し、Python へ CARLA ライブラリをインストールしてください。

```
>pip3 install --user c:%carla%carla%PythonAPI%carla%dist%carla-0.9.13-cp37-cp37m-win_amd64.whl
```

⑦ UnrealEngine へ RoadRunner インポートプラグインのインストール

<https://jp.mathworks.com/help/roadrunner/ug/downloading-plugins.html>

を開き、RoadRunner インポートプラグインをダウンロードしてください。



解凍したフォルダ内の、[.¥RoadRunner Plugins 1.3.1¥Unreal¥Plugins]より全てのプラグインを
c:¥carla¥carla¥Unreal¥CarlaUE4¥Plugins 内にコピーしてください。

・プラグインのリビルド

・コマンドプロンプトを開き、以下のコマンドを実行し、ファイルをダウンロードしてください。

```
>cd c:¥carla¥carla
```

```
>make launch
```

正常に完了すると、CARLA が起動します。

4.4.4 自動運転制御モデル用マップのインポート

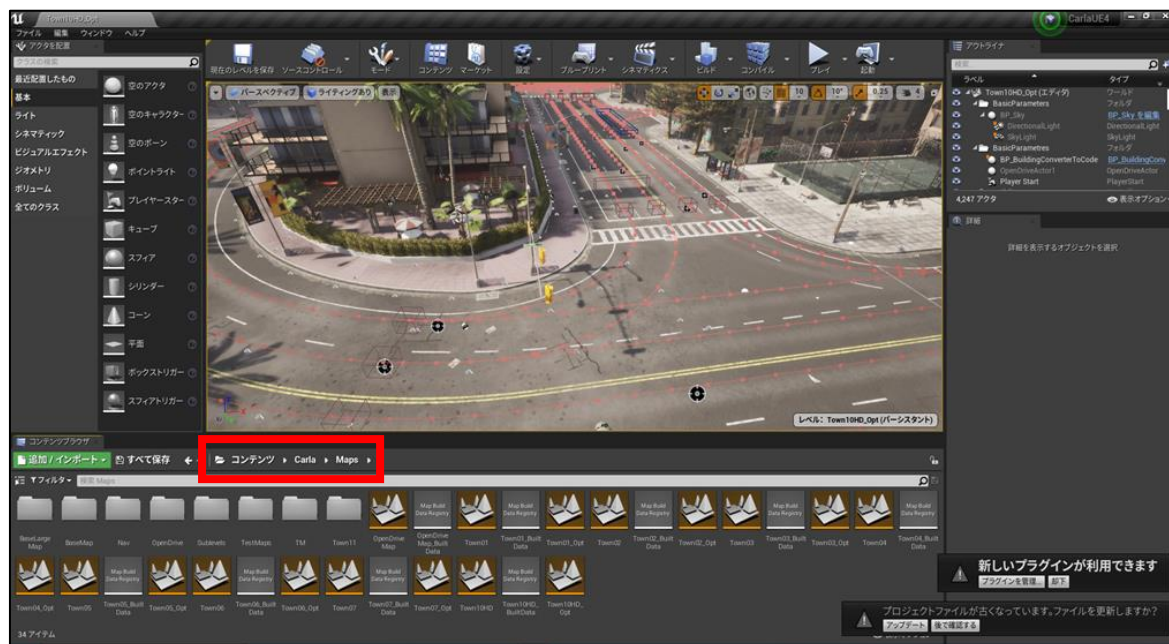
① CARLA の起動

c:¥carla¥carla¥Unreal¥CarlaUE4¥CarlaUE4.uproject をダブルクリックしてください。

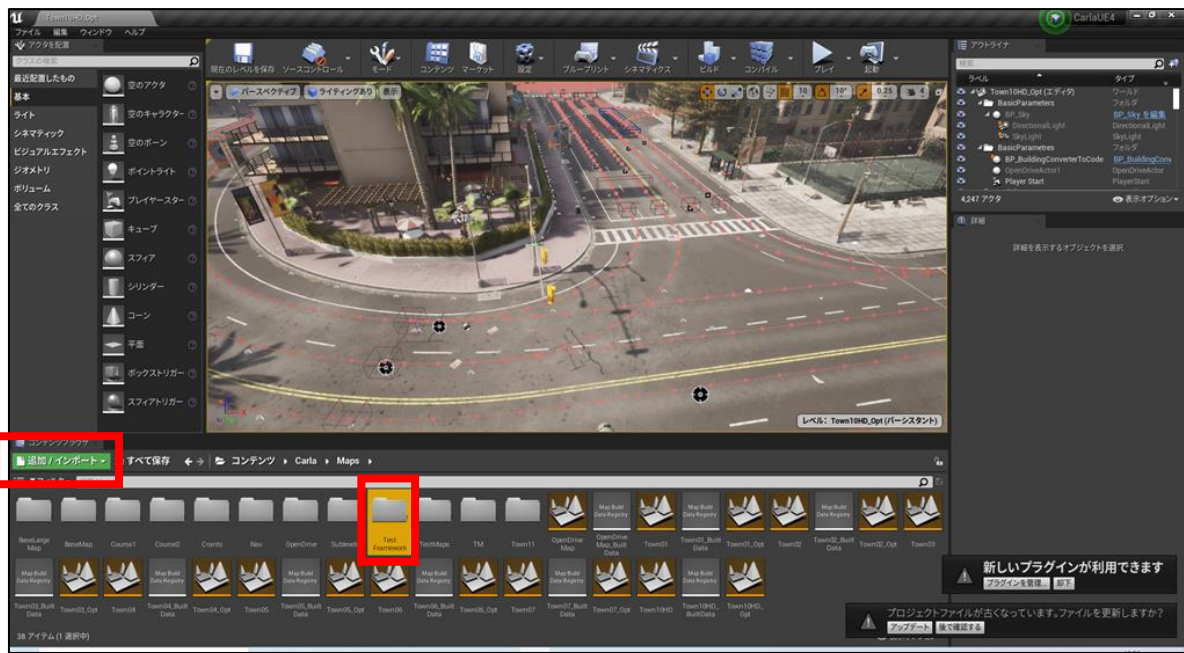
② マップのインポート(ダウンロードした自動運転制御モデルフォルダ内の、[Maps]フォルダのデータを利用)

以下を、マップ名[crants] [TestFramework] [Course1][Course2]についてそれぞれ実施してください。

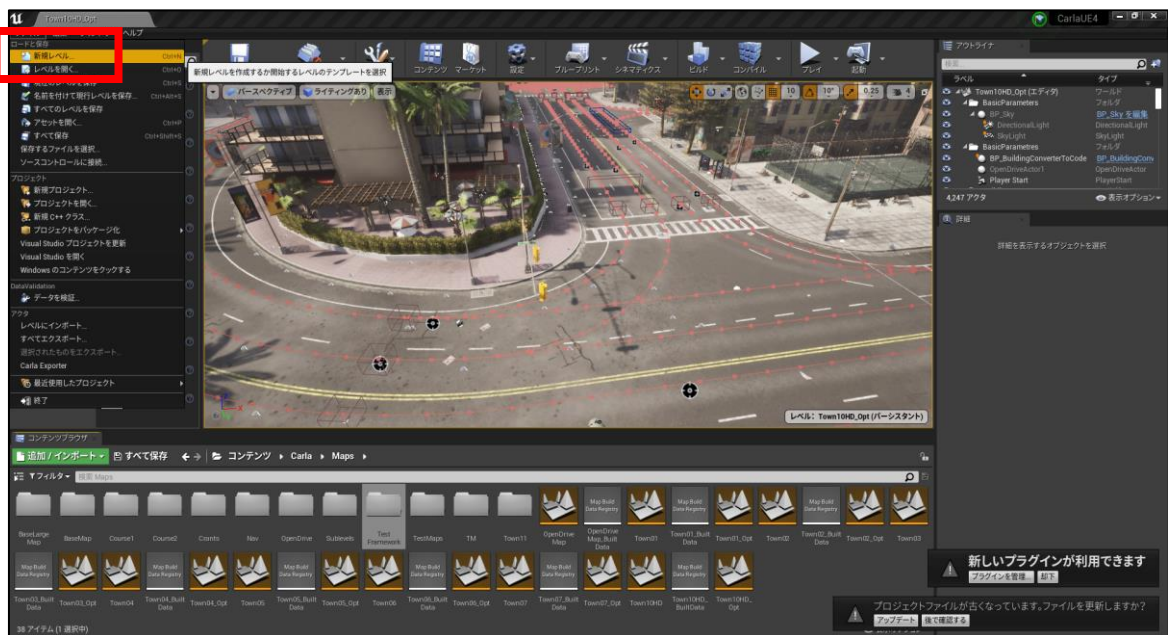
・UnrealEngine 画面の下部で、コンテンツ> Carla>Maps を開いてください。

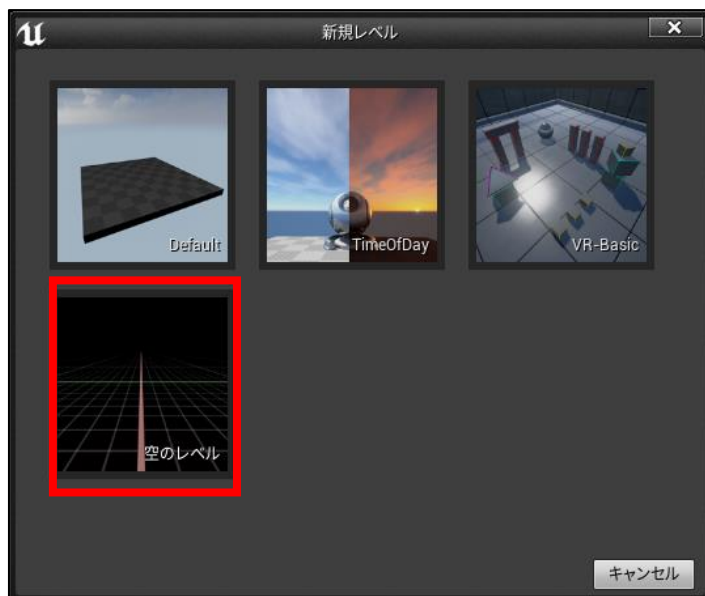


- ・[追加/インポート]をクリックし、[新規フォルダ]をクリックし、[上記マップ名]を入力しフォルダを作成してください。

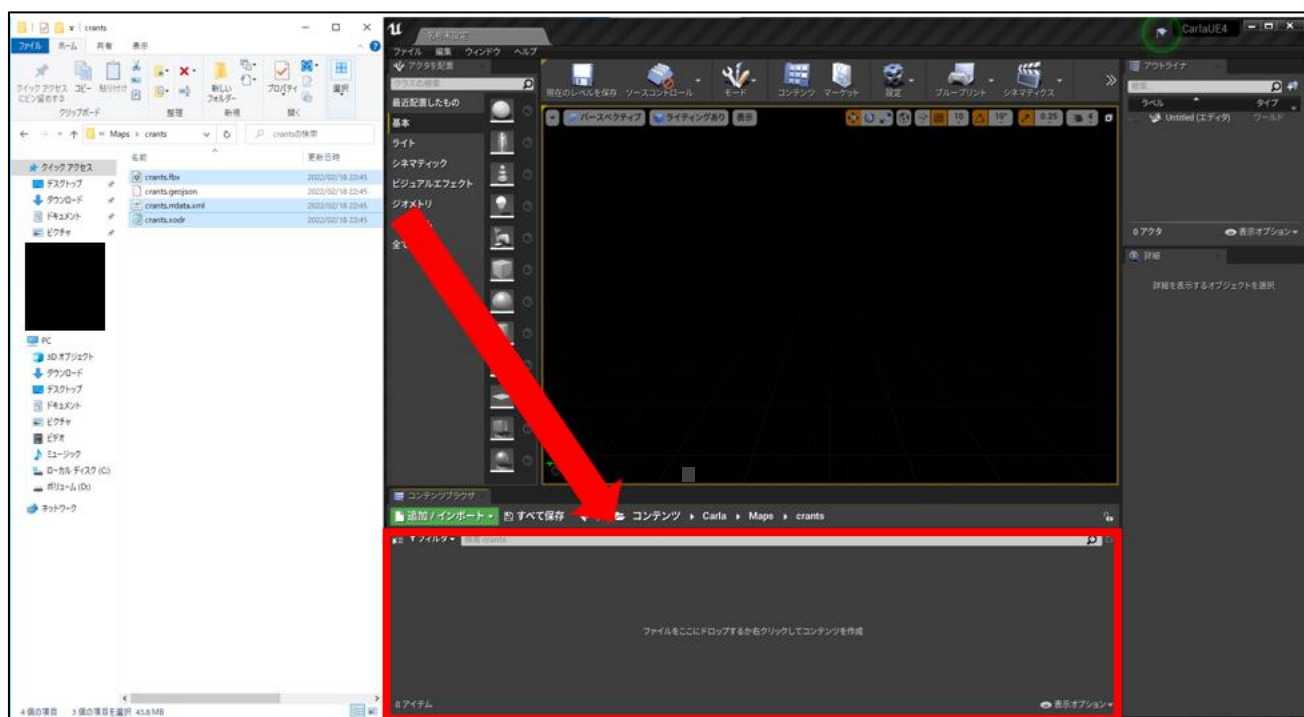


- ・メニューバーの[ファイル]より、[新規レベル]をクリックし、[空のレベル]を作成してください。

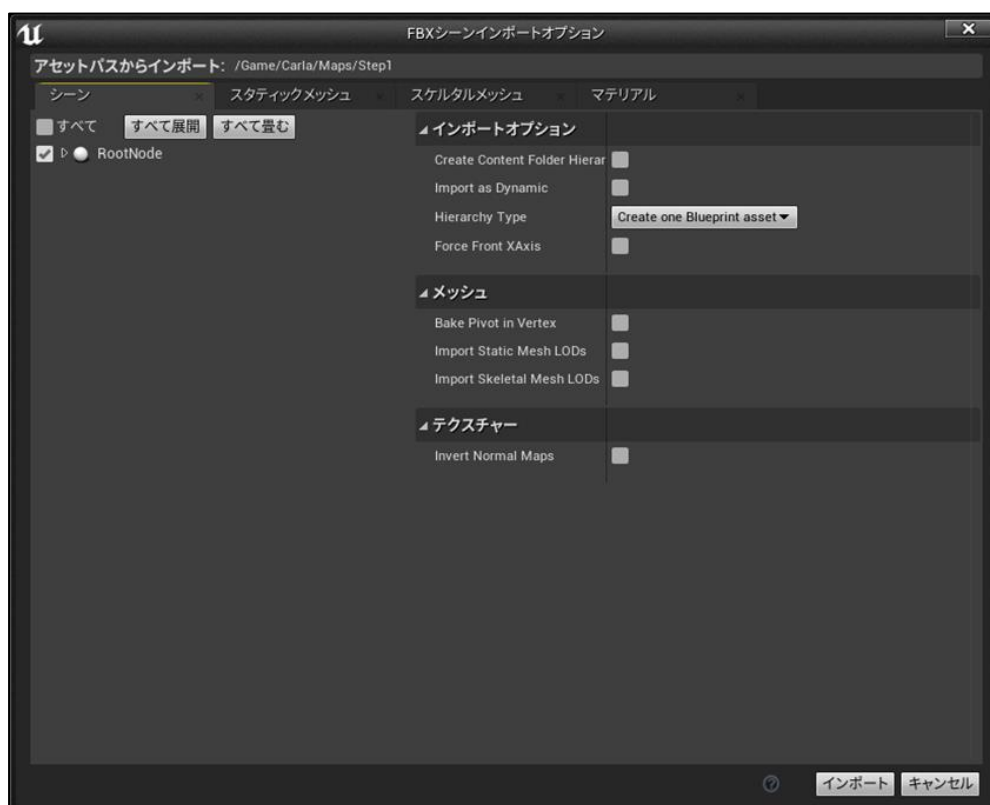
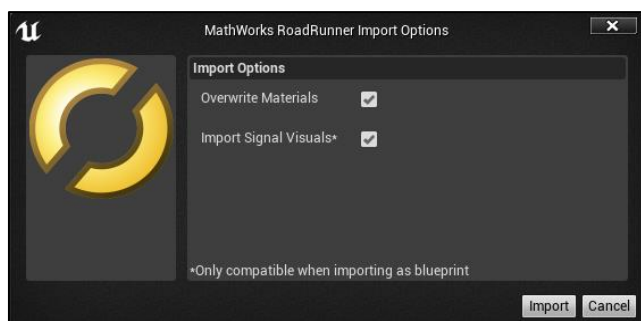


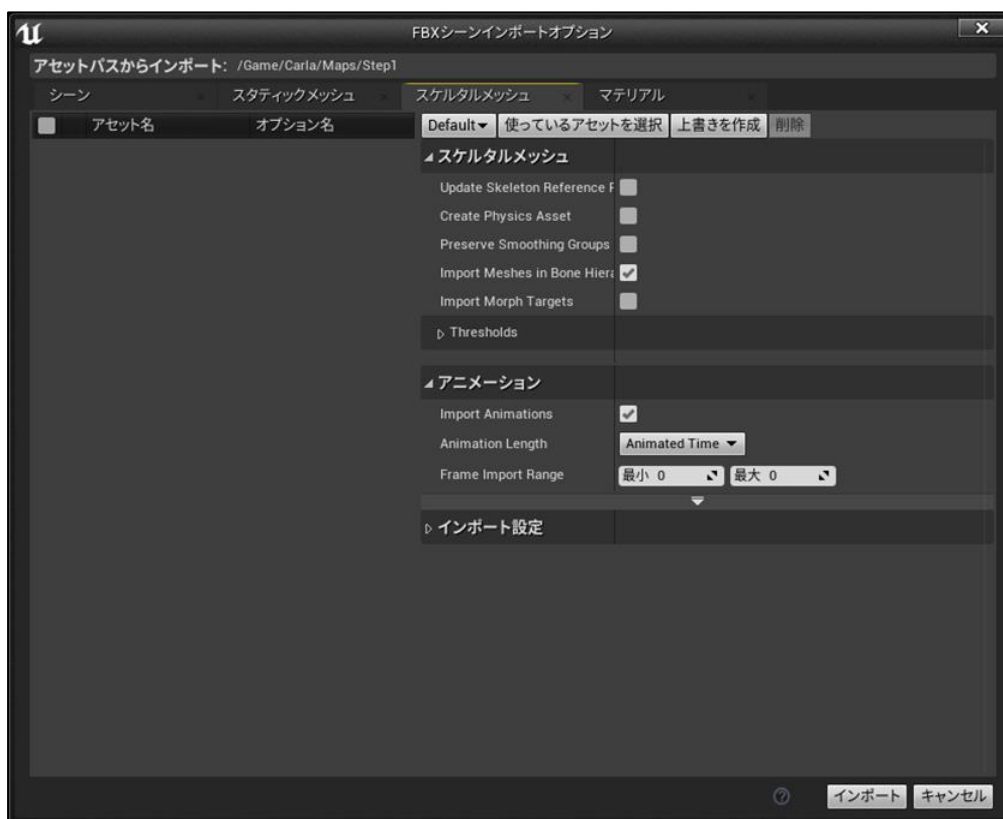
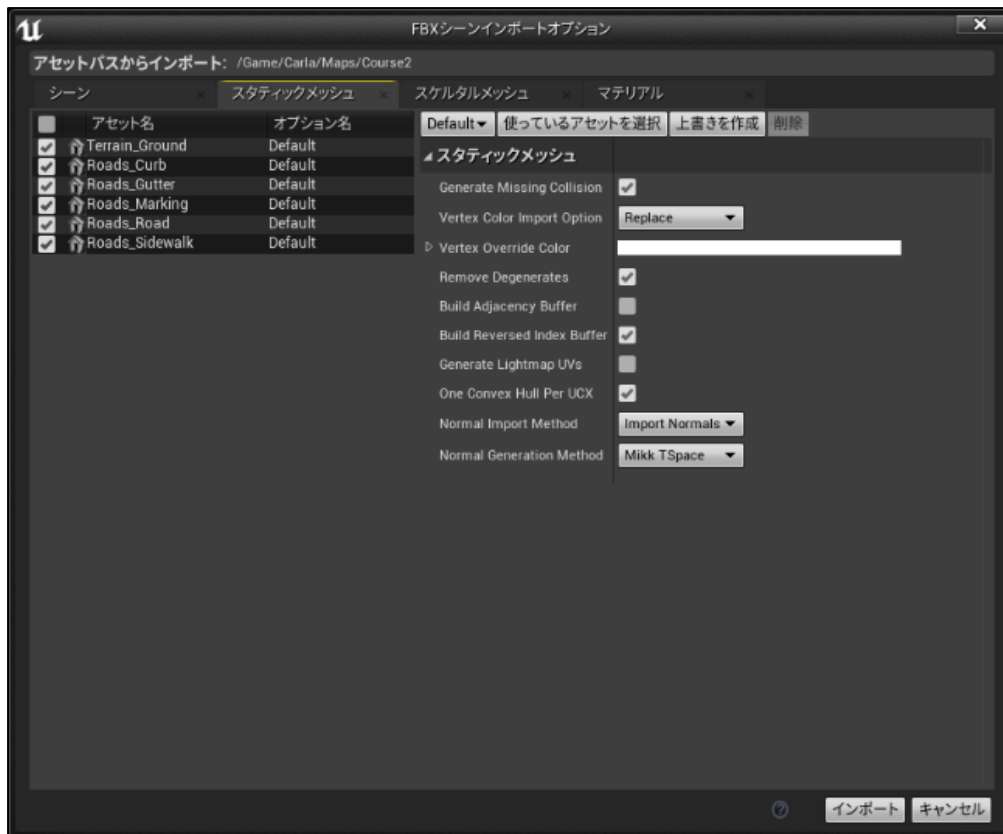


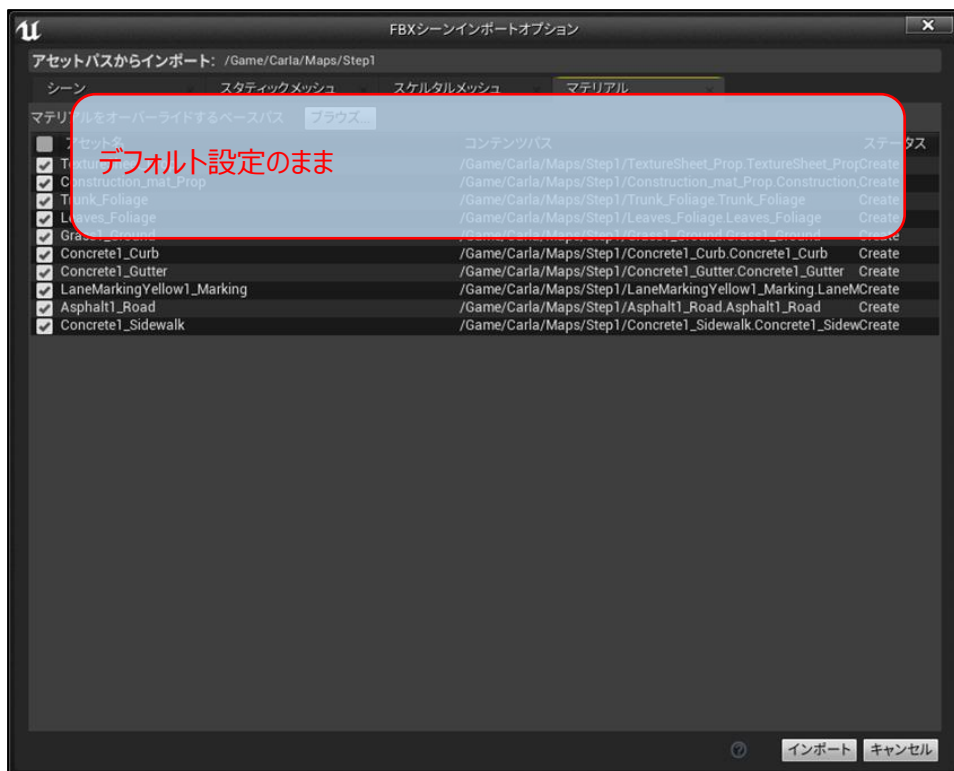
- ・ダウンロードしたファイル内の自動運転制御モデルフォルダ内の[Maps]フォルダより、[*.*fbx][*.*xml][*.*xodr]の拡張子を持つファイルを選択し、UnrealEngine の[上記マップ名]フォルダの中にドラッグアンドドロップしてください。



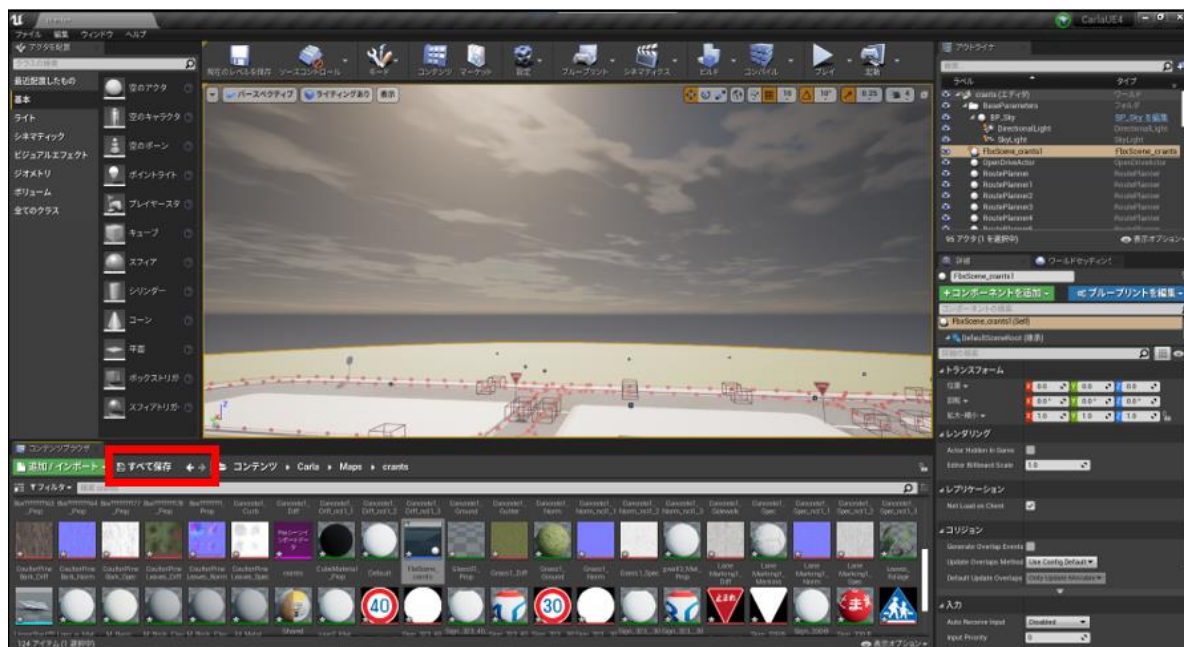
・インポートプラグインが起動後、以下の設定でインポートを実行してください。

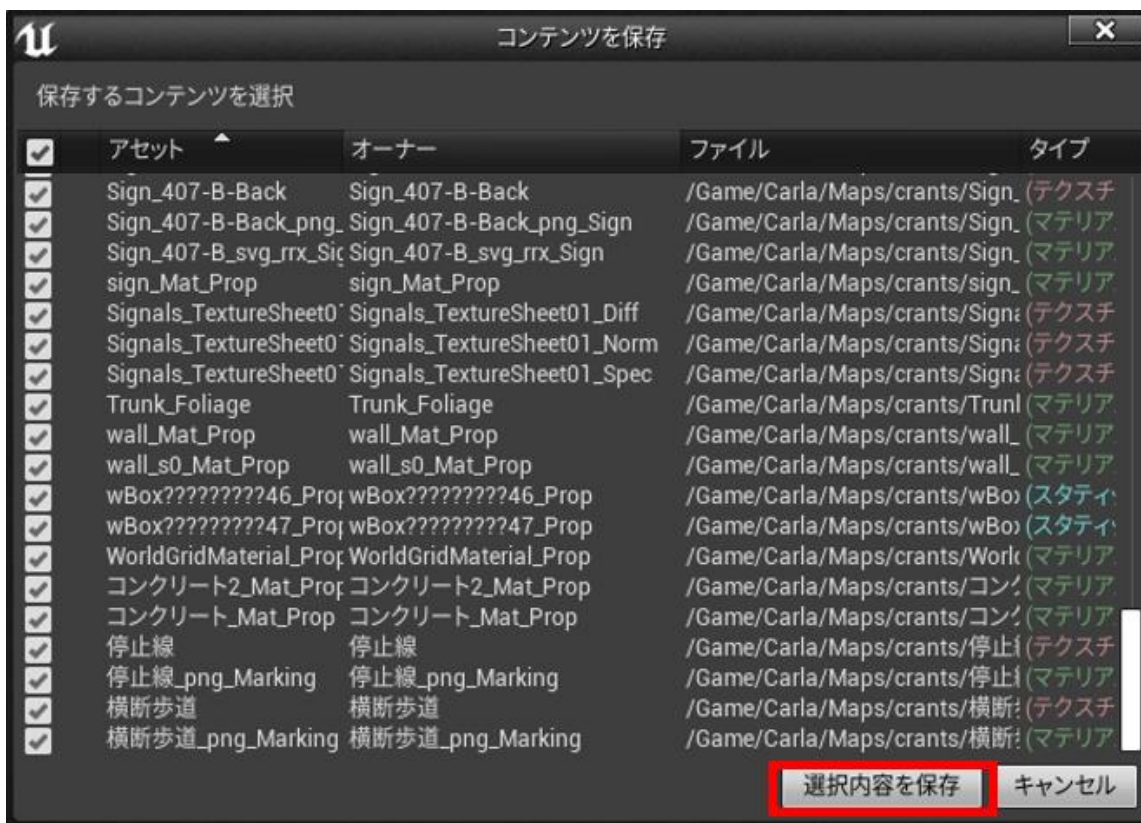




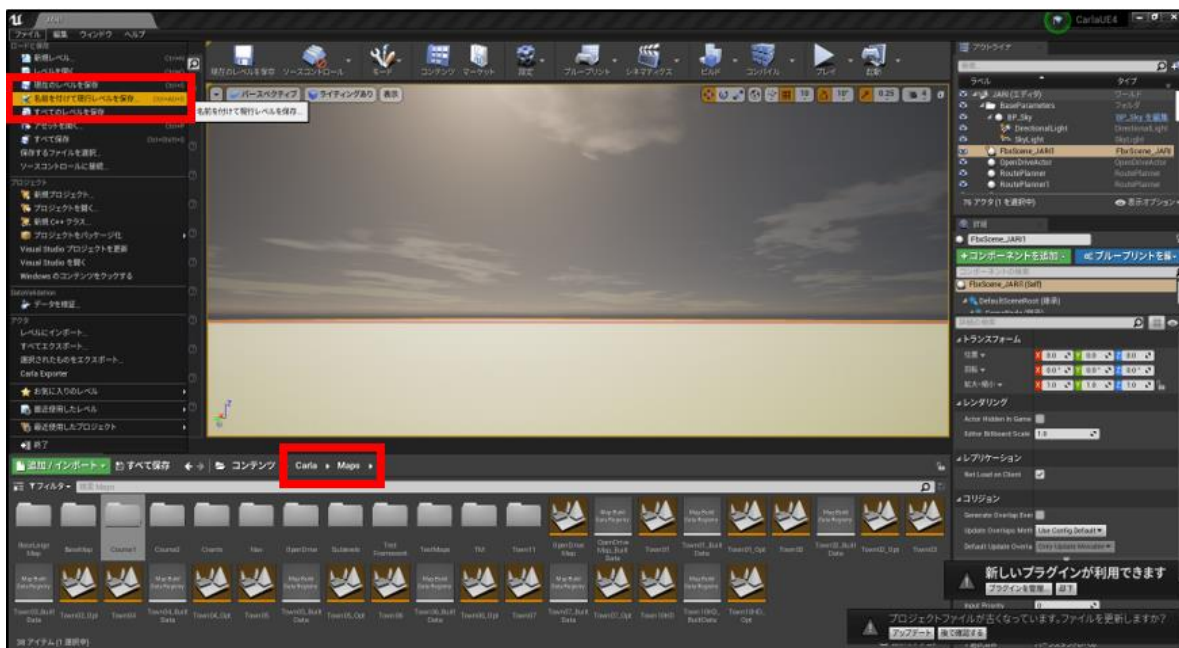


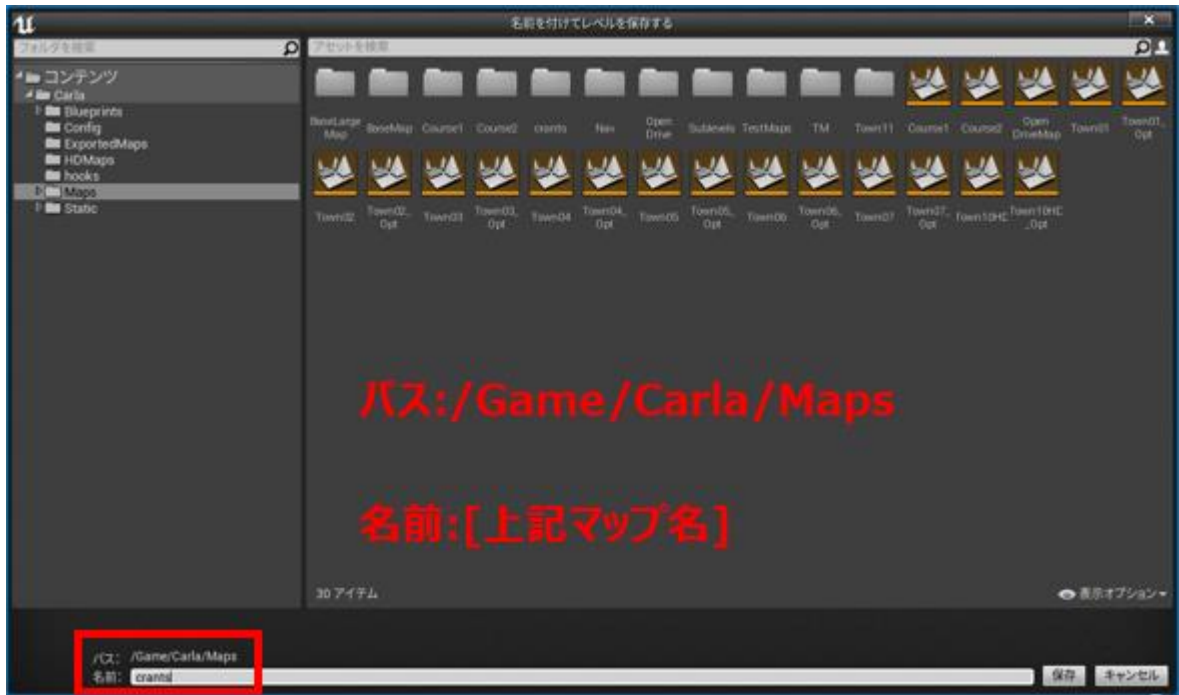
・コンテンツブラウザの[すべて保存]をクリックし、コンテンツを保存してください。





・[ファイル][名前を付けて現行レベルを保存]をクリックし、Carla#Maps フォルダに[上記マップ名]で保存してください。





・ダウンロードした“自動運転制御モデル IF ガイドライン準拠ジェネリックモデルシミュレータ環境”フォルダ内の、[Carla ビルド用追加ファイル¥Carla ビルド用追加ファイル.zip]内のファイルを、[c:¥carla¥]内へ上書きコピーしてください。

- ③ シミュレーションの実行
5 シミュレーションの実行手順へ進んでください。

5 シミュレーションの実行手順

シミュレーション形式は、パソコンの性能に合わせる為、以下の 3 パターンを用意しております。状況に合わせて選択してシミュレーションを実行してください。

- ①フルスペック：全機能を使用します。最も負荷の高いシミュレーションとなります。
 - ②物体認識なし：比較的負荷の高い認識関連のプログラムを無効にし、真値を利用したシミュレーションとなります。
 - ③仮想環境なし：認識関連及び、仮想環境を使用しません。最も負荷の軽いシミュレーションとなります。
- そのため、CARLA の実行は不要です。

5.1 基本操作

シミュレーションを実行し、結果を確認するまでの基本的な流れは以下の通りです。

5.1.1 CARLA を起動する

※仮想環境を構築していない場合は手順 5.1.2 に進んでください。

〈仮想環境を手順 4.2,4.3 で構築した場合〉

c:¥carla¥WindowsNoEditor¥CarlaUE4.exe をダブルクリックしてください。

〈仮想環境を手順 4.4 で構築した場合〉

c:¥carla¥carla¥Unreal¥CarlaUE4¥CarlaUE4.uproject をダブルクリックしてください。

UnrealEngine の「プレイ」ボタンを押し、仮想環境を実行してください。

画面上でマウスの左ボタンを押しながらドラッグすると、カメラの向きを変えることができます。

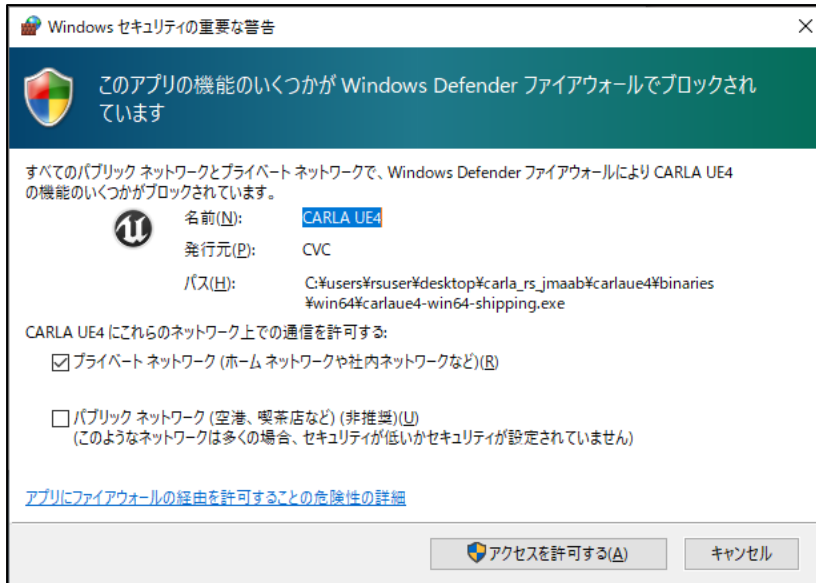
キーボードの、W,S,A,D ボタンで、カメラの位置を前後左右に移動することができます。



※SmartScreen の警告画面が出た場合は、実行を許可してください。

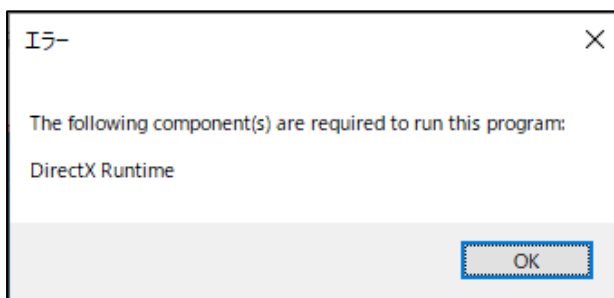


※ファイアウォールの警告画面が出た場合は、接続を許可してください。



※DirectXRuntime が必要との画面が出た場合は、以下の URL よりインストールしてください。

<https://www.microsoft.com/ja-jp/download/details.aspx?id=35>



※VisualC++ランタイムが必要との画面が出た場合は、以下の URL から、

VisualStudio2015,2017,2019,2022 の再頒布パッケージ x64 をインストールしてください。

<https://docs.microsoft.com/ja-jp/cpp/windows/latest-supported-vc-redist?view=msvc-170#visual-studio-2015-2017-2019-and-2022>

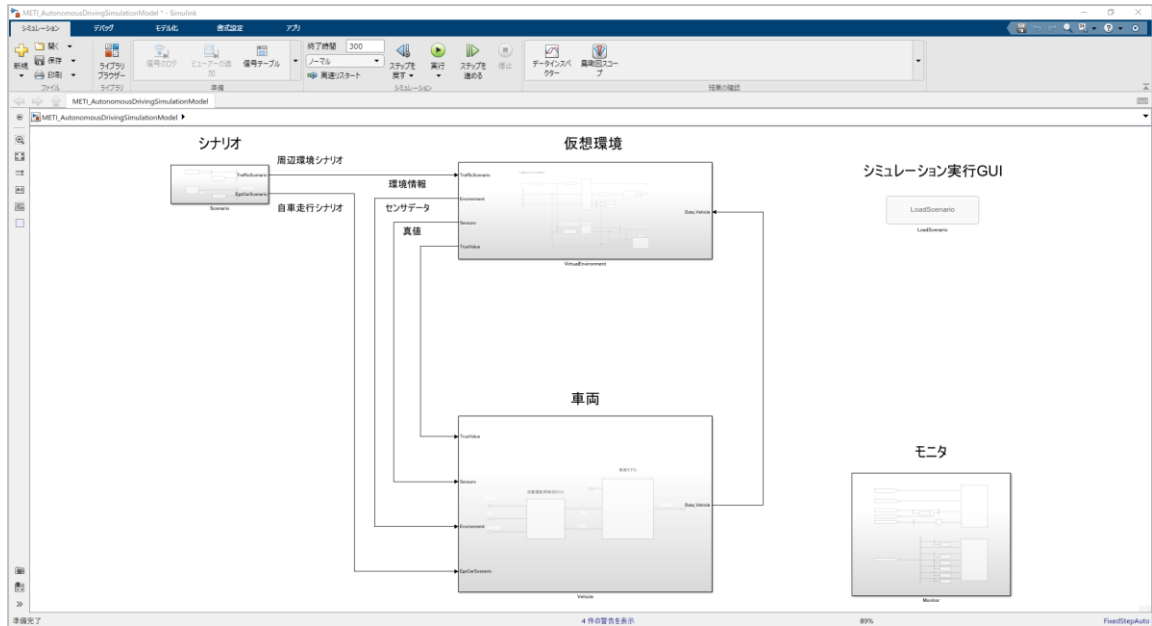
5.1.2 シミュレーションを実行する

① init_setting.m の実行

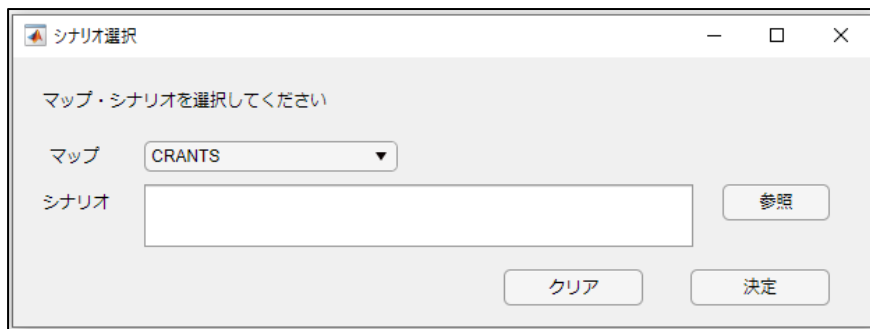
MATLAB を起動し、「現在のフォルダー」を自動運転制御モデルが存在するフォルダに変更してください。

init_setting.m を実行してください。

自動運転制御モデルとシナリオ選択ウィンドウが開きます。

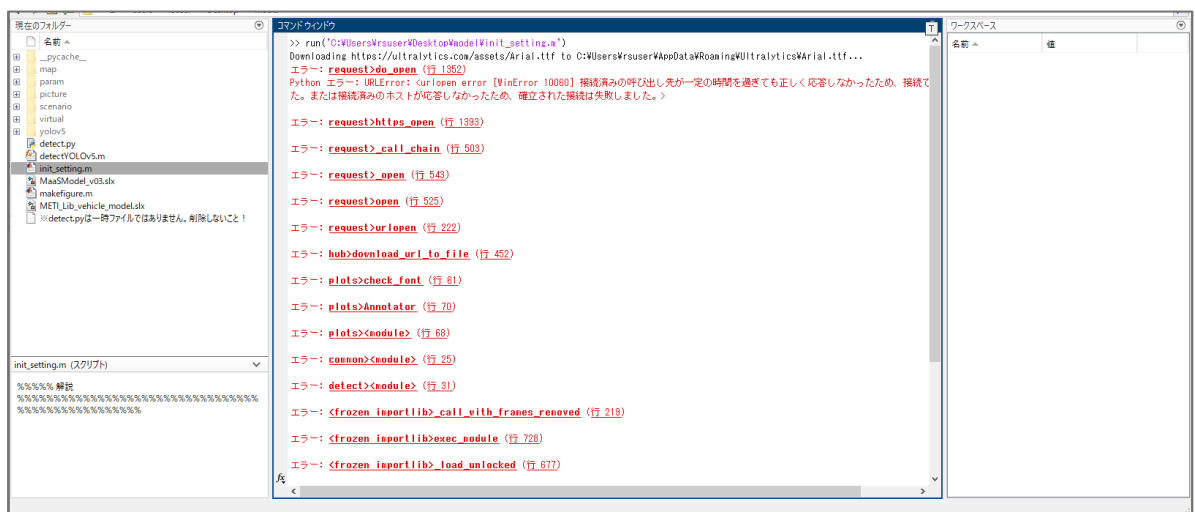


<自動運転制御モデル(METI_AutonomousDrivingSimulationModel)>



<シナリオ選択ウィンドウ> ※Simulink 上の「LoadScenario」ボタンからも起動できます。

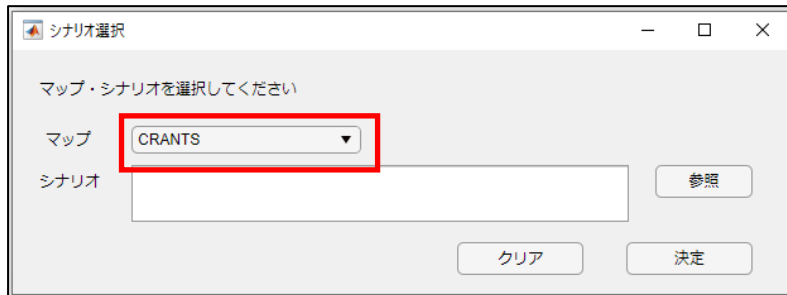
※以下のようにエラーが出る場合は、インターネット接続を確認してください。初回のみ接続が必要です。



② マップを選択

シナリオ選択ウィンドウで、マップを選択してください。

マップは「CRANTS」、「JARI 城里テストコース」、「市街地テストコース」、「安全性評価フレームワーク」から選べます。



③ シナリオを選択

「参照」を押すと、ファイル選択のダイアログボックスが開きます。

マップに対応したシナリオが表示されるので、実行したいシナリオを選択してください。

※選択するシナリオファイルの拡張子は、マップによって異なります。

- ・CRANTS、JARI 城里テストコース、市街地テストコース：xosc ファイル
※フォルダ内の xosc ファイルのうち、Catalog ファイルでないものを選んでください。
- ・安全性評価フレームワーク：mat ファイル

④ 「決定」ボタンを押下

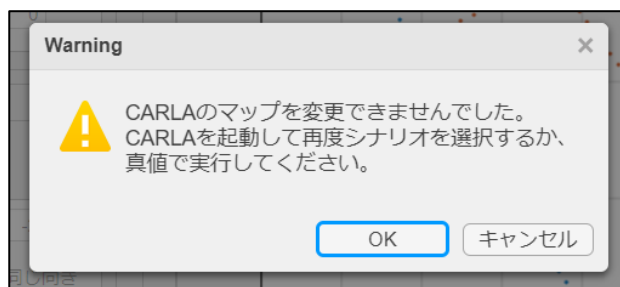
シナリオが読み込まれ、パラメータ設定画面が表示されます。

また、CARLA を起動している場合は選択したマップに自動で切り替わります。

※CARLA を起動していない等により、CARLA のマップを切り替えることができなかった場合、
下図のようなエラーが表示されます。

初回のマップ読み込み時など、マップの切り替えに時間を要している場合にも表示されます。

仮想環境で実行したい場合、CARLA を起動してからシナリオを再度選択しなおしてください。



⑤ 実行環境を選択

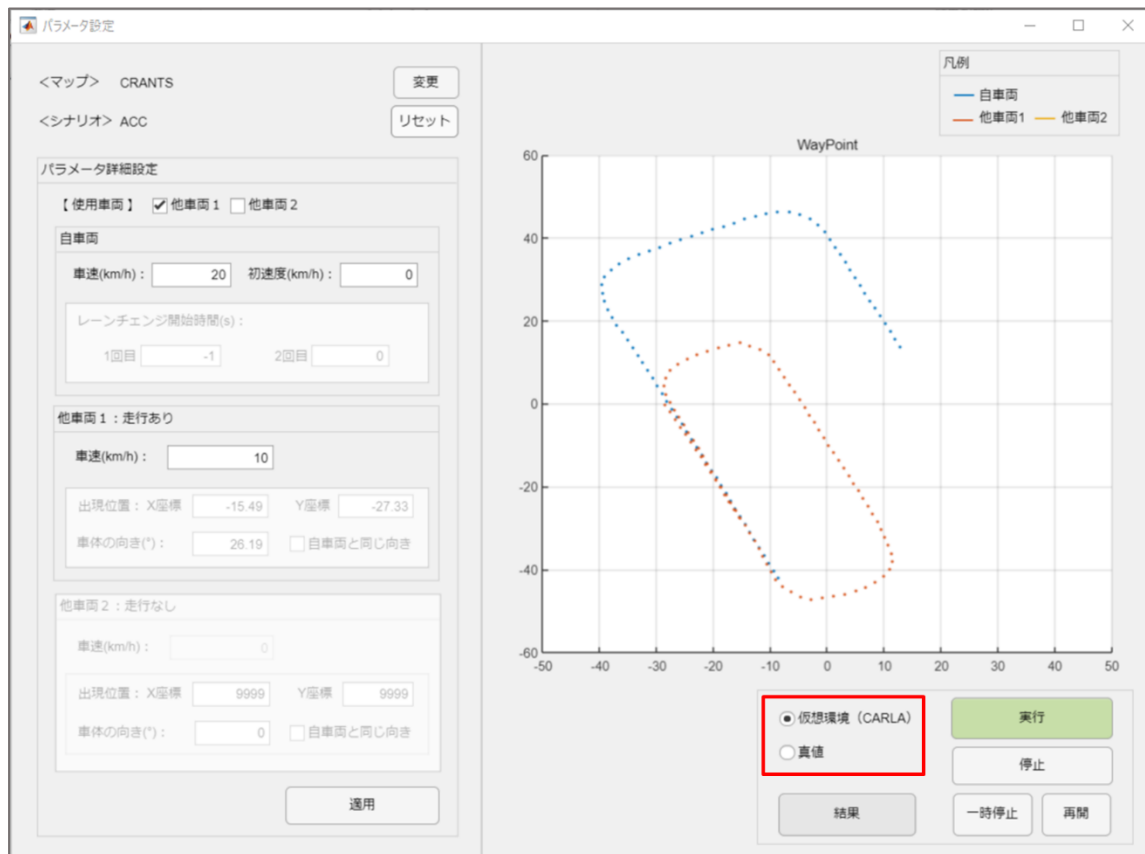
パラメータ設定ウィンドウの右下にて、シミュレーション実行環境を選択します。

- ・仮想環境（CARLA）
フルスペックでの実行。認識関連のセンサデータは仮想環境から取得されます。
CARLA を起動していないと実行できません。
- ・真値

比較的負荷の高い認識関連のプログラムを無効にし、真値を利用します。
CARLA を起動しなくても実行できます。CARLA を起動している場合、
CARLA 上で車両の動作は確認可能です。

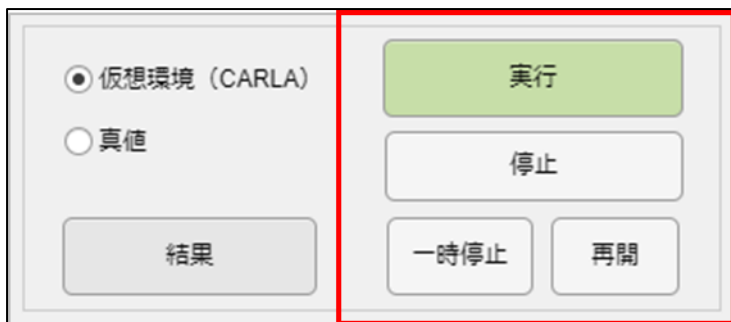
※パラメータ設定ウィンドウは、[CRANTS、JARI 城里テストコース、市街地テストコース]と[評価フレームワーク]
で外観が異なります。

※シナリオを変更したい場合は「変更」ボタンを押してください。シナリオ選択ウィンドウに戻ります。

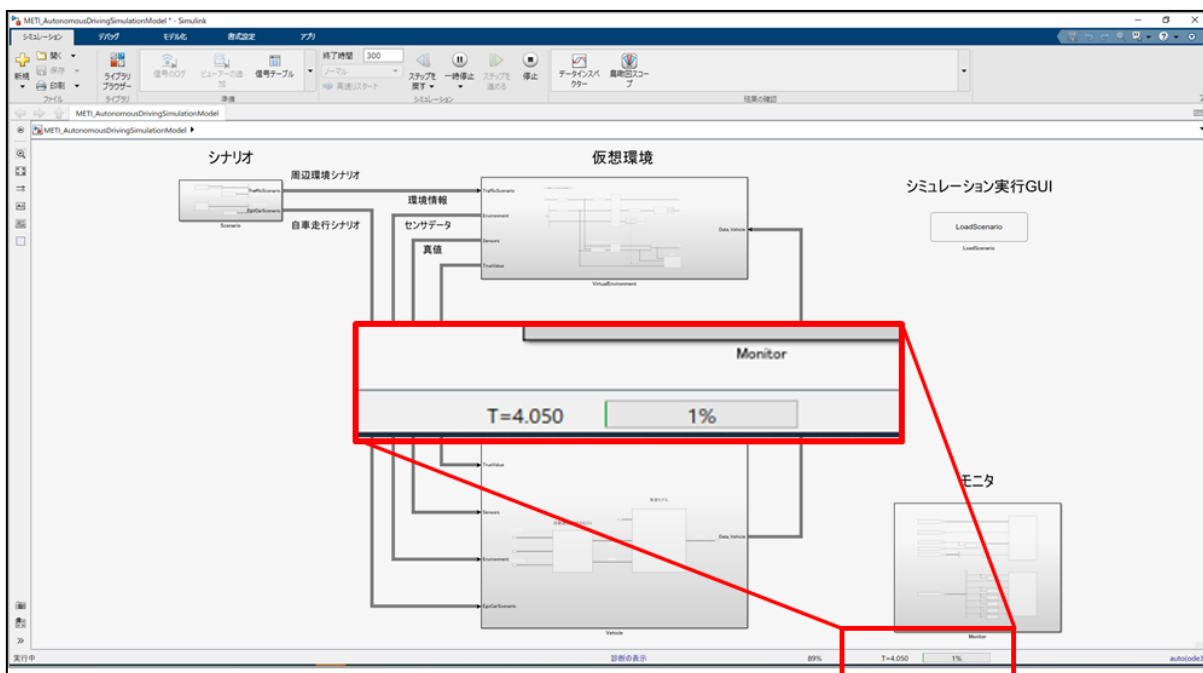


⑥ シミュレーションの実行

- ・「実行ボタン」を押すと、選択した実行環境にてシミュレーションを実行します。
※前回のシミュレーション結果は破棄されます。
- ・「停止」ボタンを押すと、実行中のシミュレーションを途中で終了します。
停止した時点までのデータは「結果」ボタンから確認できます。
- ・「一時停止」ボタンを押すと、実行中のシミュレーションを一時停止します。
一時停止をした時点までのデータは「結果」ボタンから確認できます。
- ・「再開」ボタンを押すと、「一時停止」を解除し、シミュレーションを再開します。



※実行状況と経過時間は Simulink モデルで確認できます。



※仮想環境で実行するとき

- ・RuntimeError が出る場合は、CARLA が起動していることを確認してください。
- ・目的のマップが表示されていない場合、パラメータ設定画面左上の「変更」ボタンを押し、再度マップとシナリオを選びなおしてください。

5.1.3 実行状況を確認する

シミュレーション実行中は、以下のデータを確認できます。

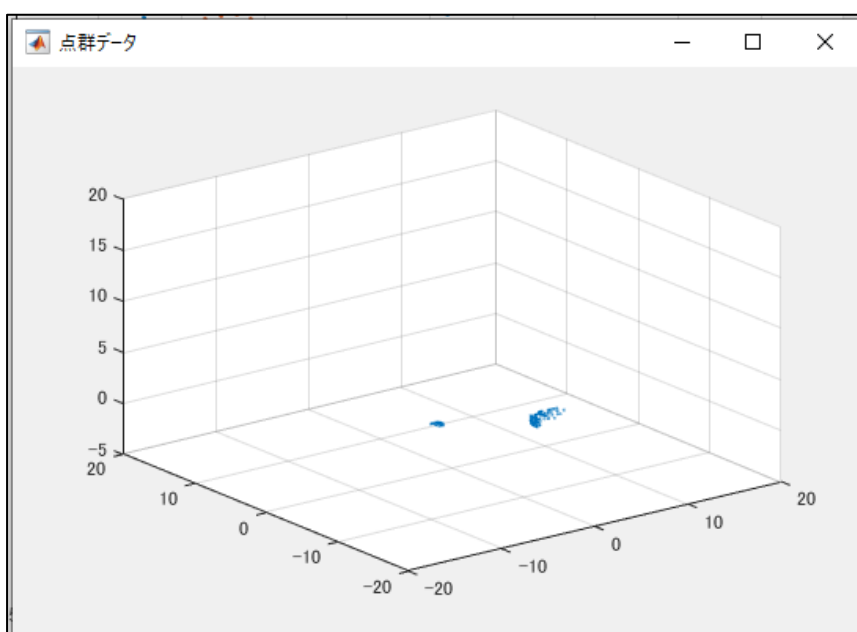
A) 自車両の車載カメラ映像(仮想環境で実行時のみ)

[自車両車載カメラ]ウィンドウにて確認できます。認識した標識は枠で囲まれます。



B) LiDAR で検出した点群データ (仮想環境で実行時のみ)

[点群データ] ウィンドウにて確認できます。

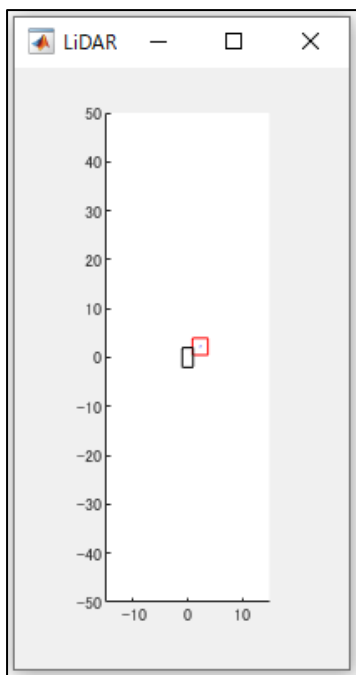


C) 障害物位置情報

[LiDAR]ウィンドウにて確認できます。

LiDAR で検出した障害物（他車両・標識など）を5つまで表示します。

※障害物は、LiDAR の点群データをクラスタリングして検出しています。



D) CARLA マップ全景 (CARLA 起動時のみ)

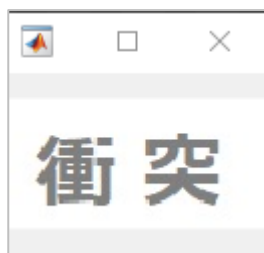
[全景]ウィンドウにて確認できます。



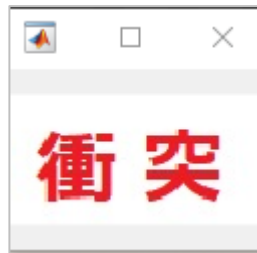
E) 衝突判定

[衝突状態]にて確認できます。衝突している間、文字が赤く表示されます。

・衝突なし

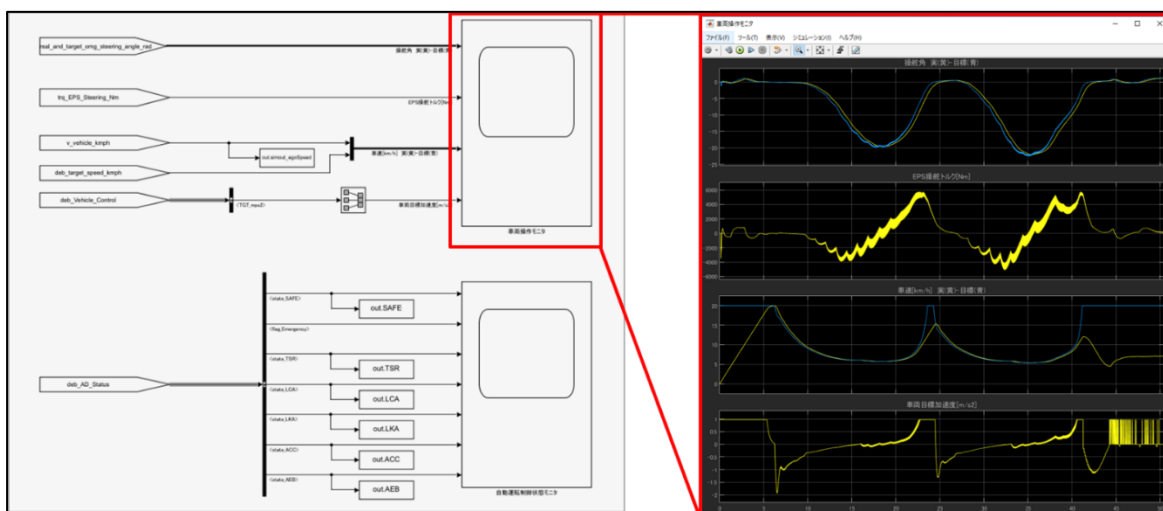
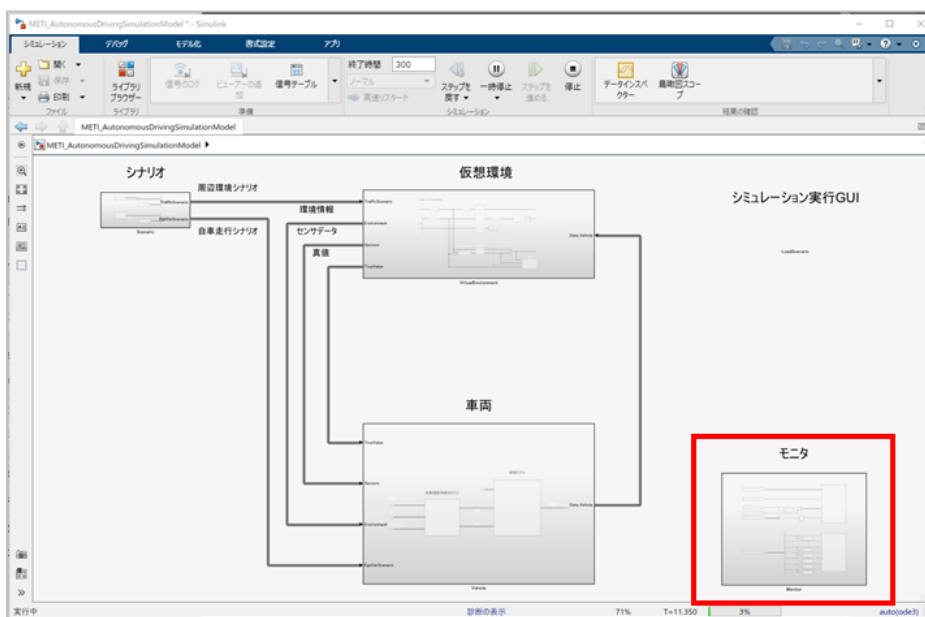


・衝突中



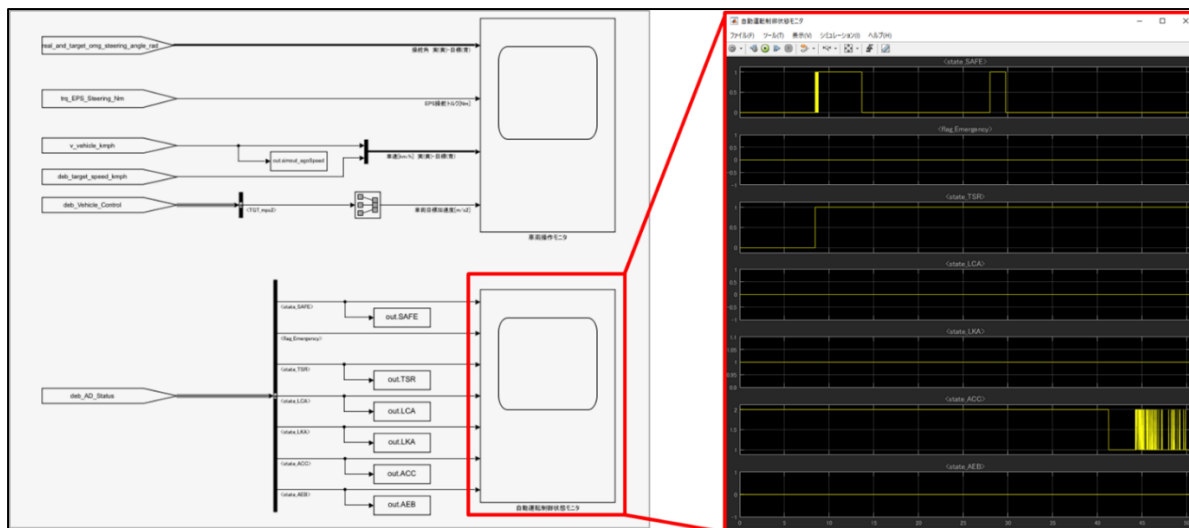
F) 自車両ステア操作状態

Simulink のモニタ内にある“ステア操作モニタ”から確認できます。



G) 自動運転制御状態 (ADAS アプリケーション)

Simulink のモニタ内にある“自動運転制御状態モニタ”から確認できます。



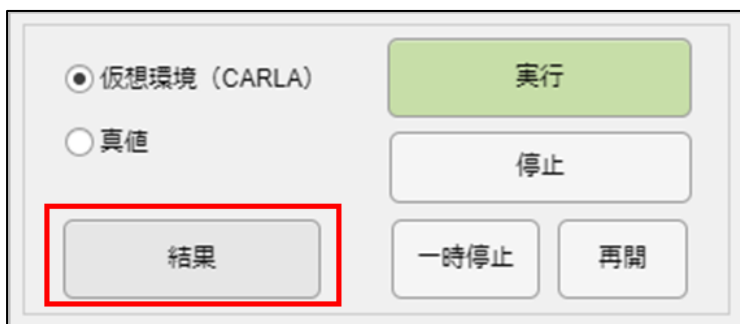
- <state_SAFE> 検出した周辺環境オブジェクトの存在位置。
- <flag_Emergency> 衝突の可能性。(0：衝突の可能性なし 1：衝突の可能性あり)
- <state_TSR> 標識の認識。(0：標識認識なし 1：標識認識あり)
- <state_LKA> レーンキープ機能。0：非作動 1：レーンキープ)
- <state_LCA> 車線変更機能。
(0：非作動 1：レーンチェンジ安全確認中 2：レーンチェンジ実行中 3：:レーンチェンジ完了)
- <state_ACC> 追従機能。(0：非作動 1：前車追従走行 2：指定速走行)
- <state_AEB> 緊急停止機能。(0：非作動 5：緩ブレーキモード 6：急ブレーキモード)

5.1.4 結果を確認する

シミュレーションを実行後、シミュレーション終了時間になる、もしくは「停止」または「一時停止」ボタンを押してシミュレーションを停止させると、実行結果が取得できるようになります。

実行結果はパラメータ設定ウィンドウの「結果」ボタンから確認できます。

※新たにシミュレーションを実行する、もしくはシナリオを変更すると、前回の実行結果は破棄されます。



結果では以下のデータが確認できます。

[車両情報]タブ

<自車両車速> 自車両の車速が時系列で表示されます。

<走行経路> 各車両の走行経路。停止車両は「*」マークで表示されます。

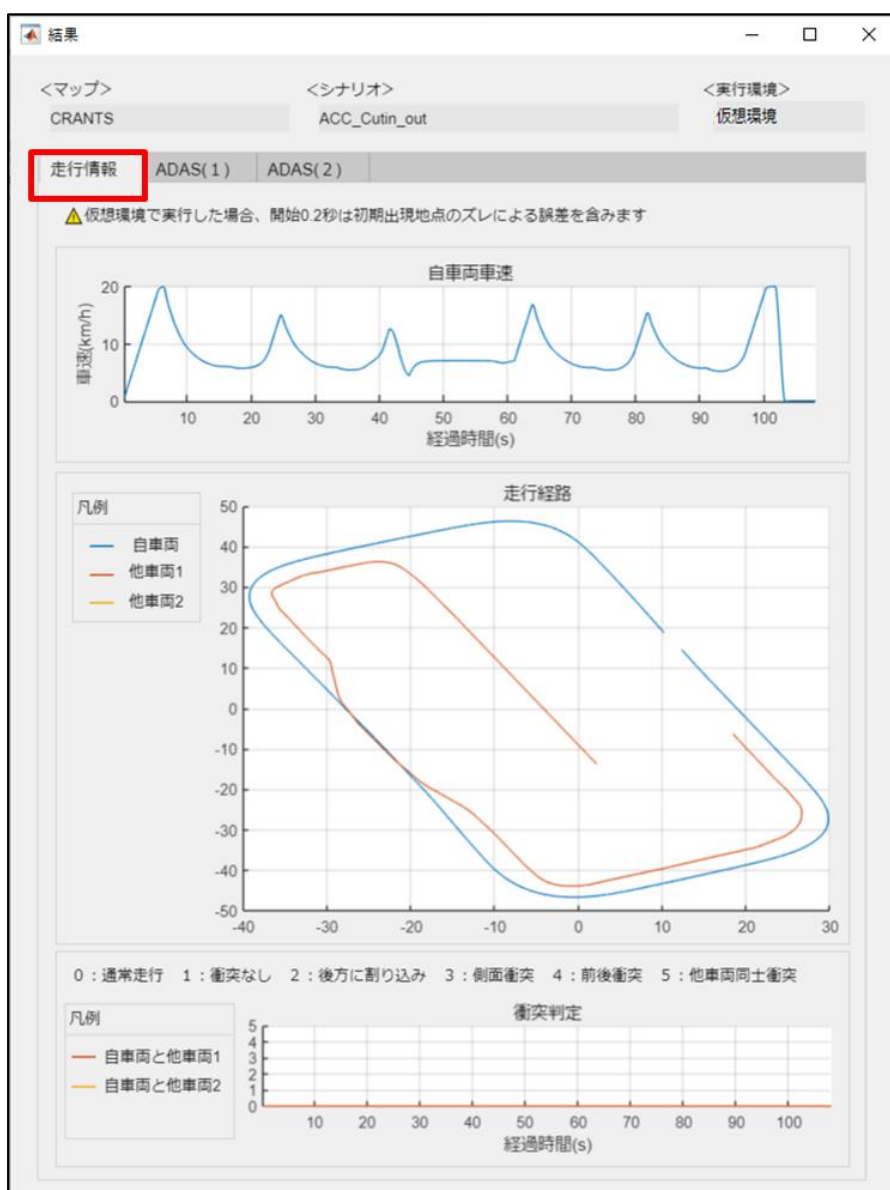
<衝突判定> 車両同士の衝突結果が表示されます。

※判定結果[2 : 後方に割り込み]については、割り込みが発生していない場合でも、他車両が自車両の前方/後方に存在する場合表示されます。

※一度衝突が起こると、衝突判定結果は保持されます。そのため再度衝突が起こらない状況に戻ったとしても、一度判定結果[3 : 側面衝突] および [4 : 前後衝突]、[5 : 他車両同士衝突]は時系列で変化しなくなります。

※他車両同士の衝突結果は、安全性評価フレームワーク No2 シナリオでのみ取得されます。

また仮想環境使用時、自車両と他車両 1/他車両 2 の衝突は仮想環境の情報から取得しますが、他車両同士の衝突は真値となります。



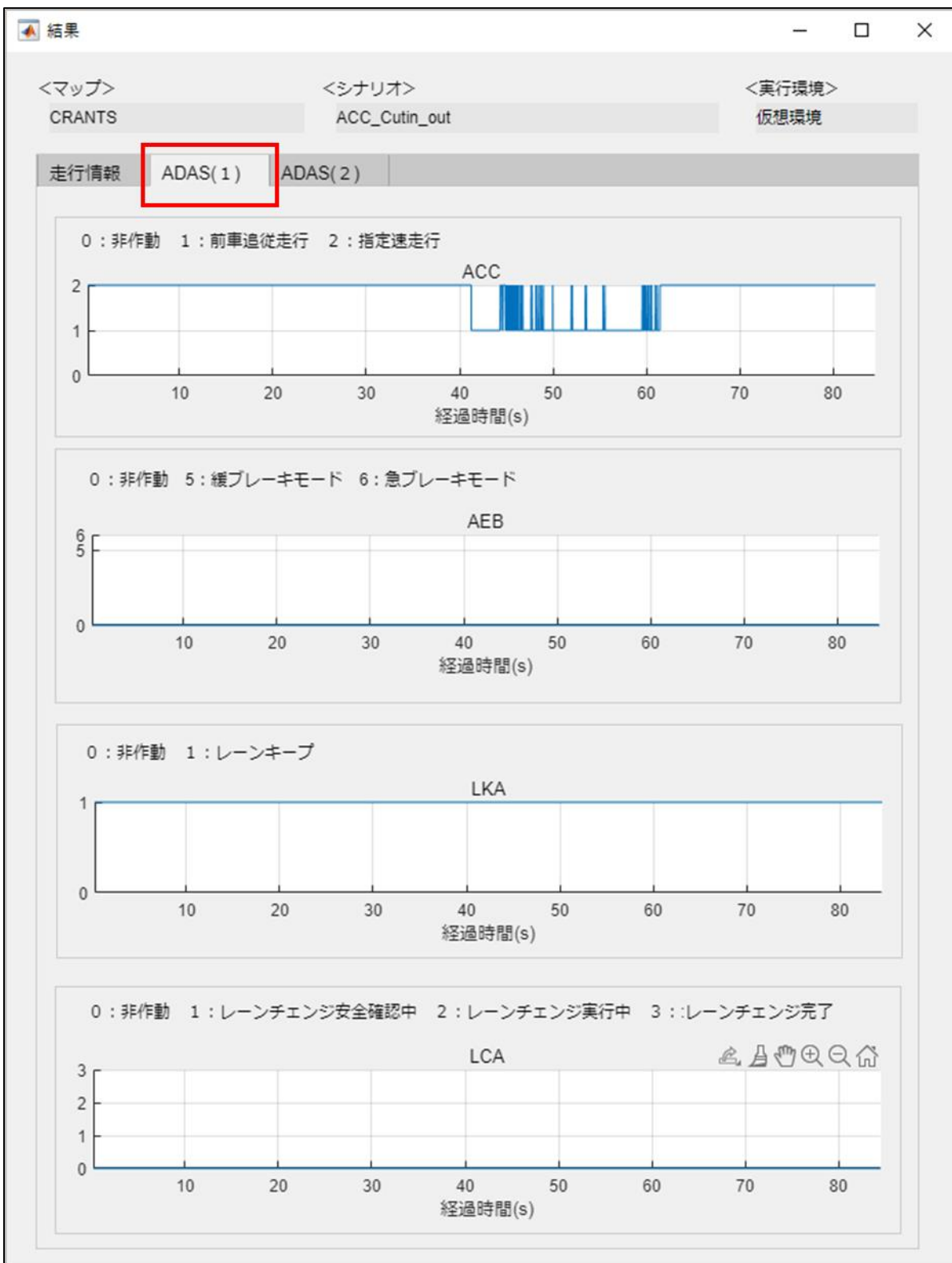
[ADAS(1)]タブ

<ACC> ACC 機能状態が時系列で表示されます。

<AEB> AEB 機能状態が時系列で表示されます。

<LKA> LKA 機能状態が時系列で表示されます。

<LCA> LCA 機能状態が時系列で表示されます。



[ADAS(2)]タブ

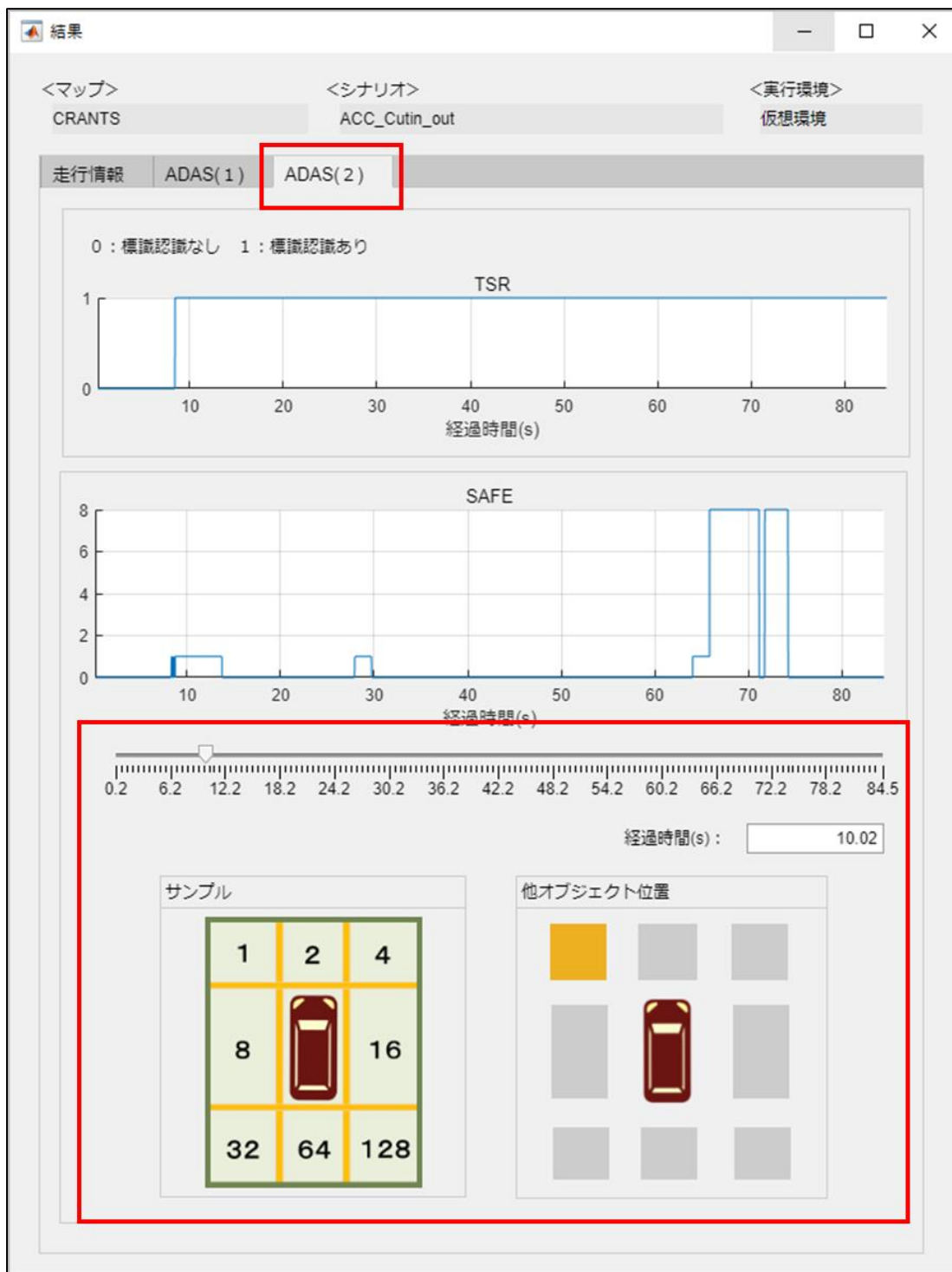
<TSR> TSR 機能状態が時系列で表示されます。

<SAFE> 認識した他オブジェクト（他車両や標識など）の存在位置情報が、時系列で表示されます。

※SAFE は、自車両が認識したオブジェクトの存在位置を数値で表したデータです。

下図の“サンプル”のように、自車両に対する各領域に数値が割り振られており、オブジェクトが存在する領域内の数値の和が出力されます。

※スライダーまたはテキストボックスで経過時間を指定すると、“他オブジェクト位置”にて存在位置を図示します。



5.2 シナリオとパラメータ

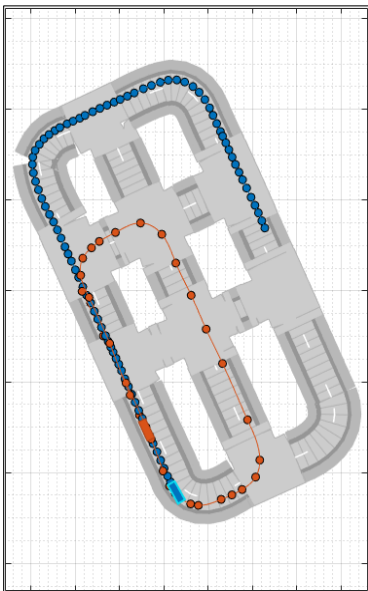
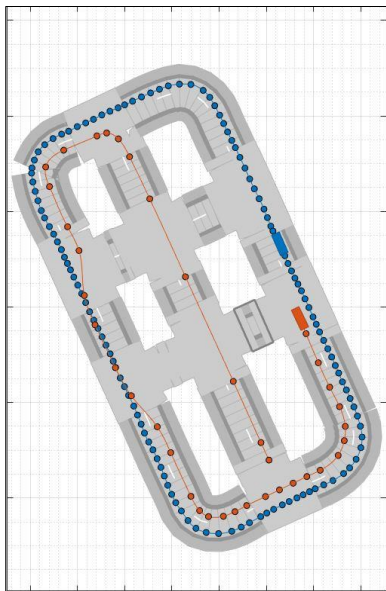
シナリオのシミュレーションを実行する前に、一部のパラメータを再設定することができます。ここで再設定できるパラメータは、マップ・シナリオごとに異なります。以下、マップごとのシナリオの詳細と、パラメータを再設定してシミュレーションを行う手順を示します。

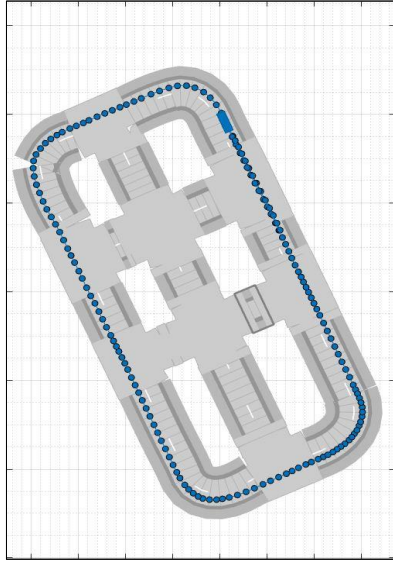
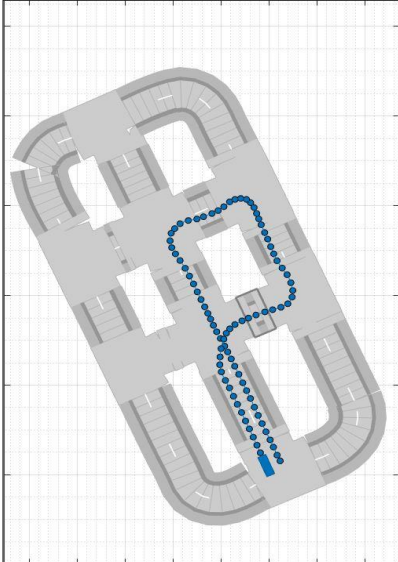
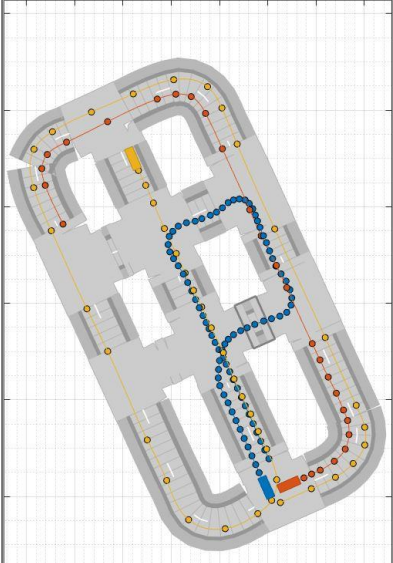
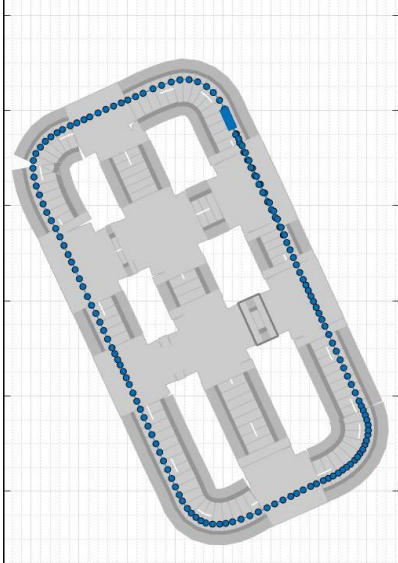
5.2.1 CRANTS・JARI 城里テストコース・市街地テストコース

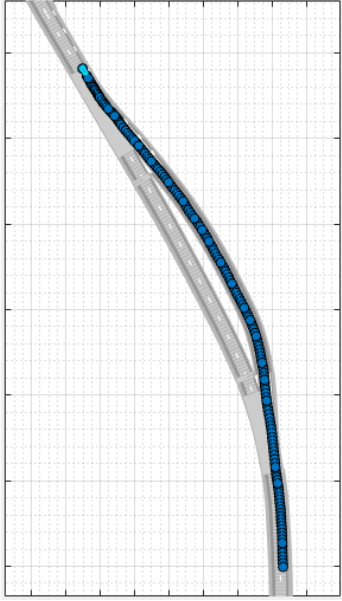
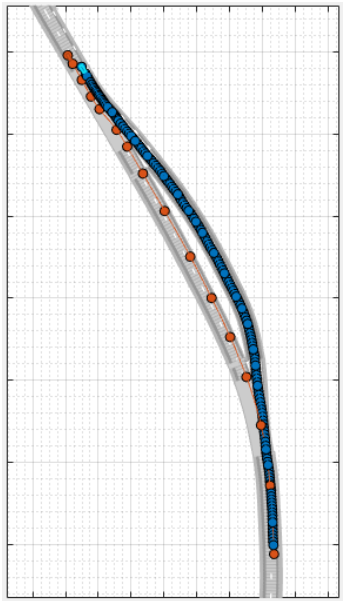
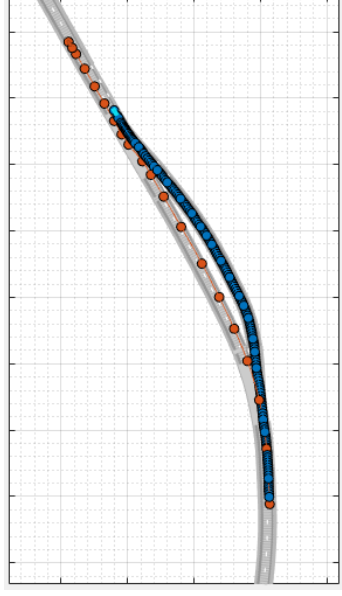
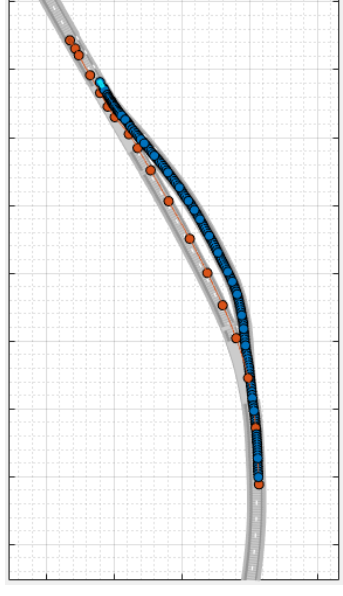
5.2.1.1 シナリオ一覧

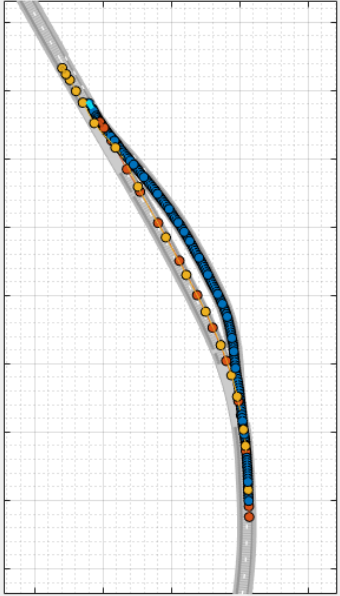
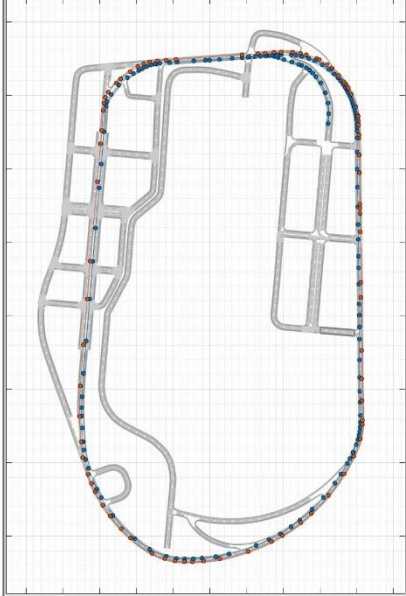
CRANTS、JARI 城里テストコース、市街地テストコースのシナリオの一覧は以下の通りです。

走行経路は、各車両の WayPoint を示しています。

マップ	シナリオ名	ACC.xosc	ACC_Cutin_out.xosc
	説明	他車両に追従する	カットインしてきた他車両に追従する
CRANTS	走行経路 ● : 自車両 ● : 他車両 1 ● : 他車両 2		

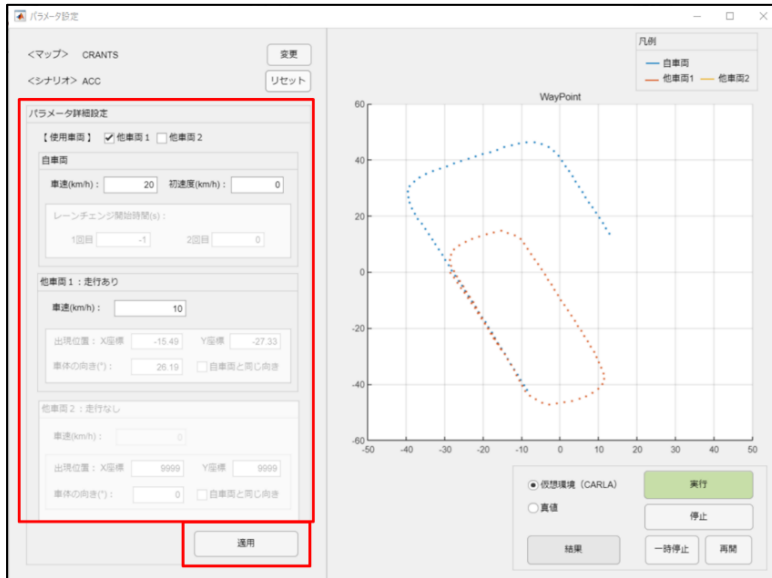
マップ	シナリオ名	Around_EgoOnly.xosc	CrossRoad_EgoOnly.xosc
CRANTS	説明	自車両 1 台で外周を走行する	自車両が 1 台で交差点及び T 字路を右折→左折→左折→左折
	走行経路 ●：自車両 ●：他車両 1 ●：他車両 2		
	シナリオ名	CrossRoad_OtherCar.xosc	LaneChange_EgoOnly.xosc
	説明	【CrossRoad_EgoOnly.xosc】のシナリオに他車両を追加し、認識しながら走行	外周を走行中に指定の時間で車線変更 ※自車両のレーンチェンジ先の経路は制御によって決まるため、下図“走行経路”のWayPointではレーンチェンジの様子は示されていません。
走行経路 ●：自車両 ●：他車両 1 ●：他車両 2			

マップ	シナリオ名	shunt_EgoOnly.xosc	shunt_OtherCar_Const_InBack.xosc
JARI 城里 テストコース	説明	自車両 1 台で合分流路を走行する	自車両が減速することなく 他車両の後方に割り込んで合流する
	走行経路 ●：自車両 ●：他車両 1 ●：他車両 2		
	シナリオ名	shunt_OtherCar_Const_InFront.xosc	shunt_OtherCar_DC_InBack.xosc
	説明	自車両が減速することなく 他車両の前方に割り込んで合流する	自車両が減速してから 他車両の後方に割り込んで合流する
走行経路 ●：自車両 ●：他車両 1 ●：他車両 2			

マップ	シナリオ名	shunt_OtherCar_ DC_InCenter.xosc	
JARI 城里 テストコース	説明	自車両が減速してから 2 台の他車両の 間に割り込んで合流する	
	走行経路 ●：自車両 ●：他車両 1 ●：他車両 2		
	市街地 テストコース	シナリオ名	Course2.xosc
説明		自車両が内周、他車両が外周を 一周走行する	
走行経路 ●：自車両 ●：他車両 1 ●：他車両 2			

5.2.1.2 パラメーター一覧

シナリオ読み込み後、パラメータ設定ウィンドウの左枠“パラメータ詳細設定”で特定のパラメータを変更できます。
 ※パラメータを変更したら、必ず「適用」のボタンを押してください。



・使用車両

表示したい他車両にチェックを入れてください。デフォルトでは、走行しない他車両は非表示となっております。

※非表示設定にした他車両の出現位置は、マップ範囲外に自動的に変更されます。

また、デフォルトの WayPoint 設定も一度削除されます。デフォルトに戻す場合は、リセットボタンを押してください。

※自車両は非表示にできません。

・自車両

自車両の目標車速と初速度を指定できます(単位 km/h)。

また、自車両がレーンチェンジを行うシナリオ【LaneChange_EgoOnly】では、レーンチェンジを開始するタイミングを時間(単位 s)で指定できます。シミュレーション開始からの経過時間を指定してください。

※レーンチェンジは 2 回まで設定できます。1 回目は初期レーンから別レーンに、2 回目は別レーンから初期レーンに戻るレーンチェンジになります。レーンチェンジを行わない場合、“-1”を指定してください。

・他車両 1 および他車両 2

他車両が走行するか否かで、設定できるパラメータが異なります。

○走行あり

他車両の車速を指定できます。(単位 km/h)

○走行なし

車両の出現位置を座標で指定できます。座標はパラメータ設定ウィンドウ右枠の“WayPoint”を参考に指定してください。また、車体の向き(単位 deg)を指定できます。

自車両の初期向きと同じ向きにしたい場合、“自車両と同じ向き”にチェックを入れると自動的に指定されます。

※走行ありの車両を走行なしにしたい場合、一度車速を“0” km/h に変更して「適用」ボタンを押すことで、走行なしになり、出現位置を指定できるようになります。デフォルトに戻す場合は「リセット」ボタンを押します。

5.2.2 安全性評価フレームワーク

一般社団法人 日本自動車工業会が公開している「自動運転の安全性評価フレームワーク Ver1.0」にて提示されている、一般車両の交通外乱シナリオの一部を簡易的に検証できます。いくつかのパラメータは変更することができ、変更可能なパラメータはカットインシナリオ、カットアウトシナリオ、減速シナリオで異なります。また、安全性評価フレームワークに従い、特定のパラメータは自動的に設定されます。

5.2.2.1 シナリオ一覧

<カットイン> No1, No5, No10, No17, No19

<カットアウト> No2, No6, No13, No20

<減速（停止）> No4, No8, No15, No22

 実装済み


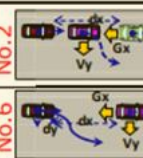

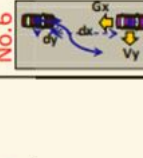




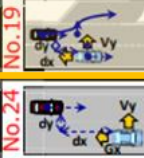
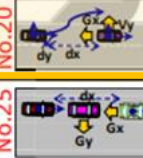
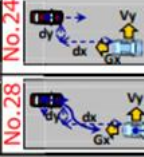
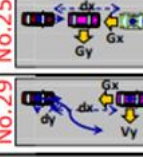

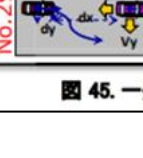
Road geometry		Ego-vehicle behavior		Surrounding vehicles position & motion					
				Cut in	Cut out	Acceleration	Deceleration (Stop)	Sync	
Main roadway	Lane keep	No.1	No.2			No.3	No.4		
	Lane change	No.5	No.6			No.7	No.8	No.9	
Merging zone	Lane keep	No.10						No.11	
	Lane change	No.12	No.13			No.14	No.15	No.16	
Departure zone	Lane keep	No.17						No.18	
	Lane change	No.19	No.20			No.21	No.22	No.23	
Ramp	Lane keep	No.24	No.25			No.26	No.27		
	Lane change	No.28	No.29			No.30	No.31	No.32	

図 45. 一般車両の交通外乱シナリオ

※一般社団法人 日本自動車工業会が公開している「自動運転の安全性評価フレームワーク Ver1.0」より抜粋

5.2.2.2 安全性評価フレームワークのマップ

直線道路 2 レーン、合流路 1 レーンおよび分流路 1 レーンを持つ計 4 レーンのマップです。

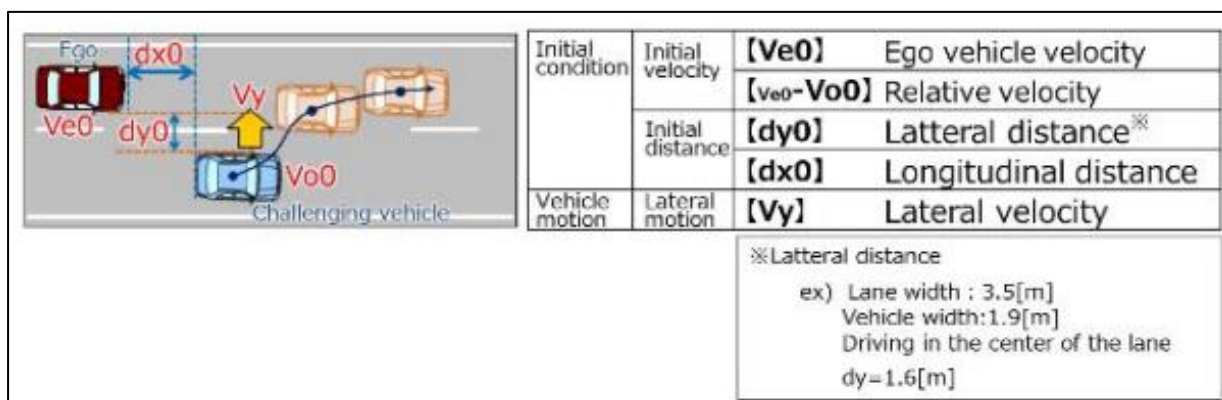
※本マップは左側走行を想定しています。そのため、右側走行を想定している安全性評価フレームワークと合流路 (Merging zone) と分流路 (Departure zone) の向きが異なります。一般車両の交通外乱シナリオの図と合流路・分流路の位置が反転していることをご承知おきください。



表 1 安全性評価フレームワークマップ レーン情報

レーン名	レーンの種類	レーンの中央(Y 座標)	進行方向の道路範囲(X 座標)	道幅(m)
MyRoute1	分流	5.25	-12 ~ 180	3.5
MyRoute2	直線	1.75	-180 ~ 180	3.5
MyRoute3	直線	-1.75	-180 ~ 180	3.5
MyRoute4	合流	-5.25	-180 ~ 12	3.5

5.2.2.3 カットインシナリオのパラメーター一覧



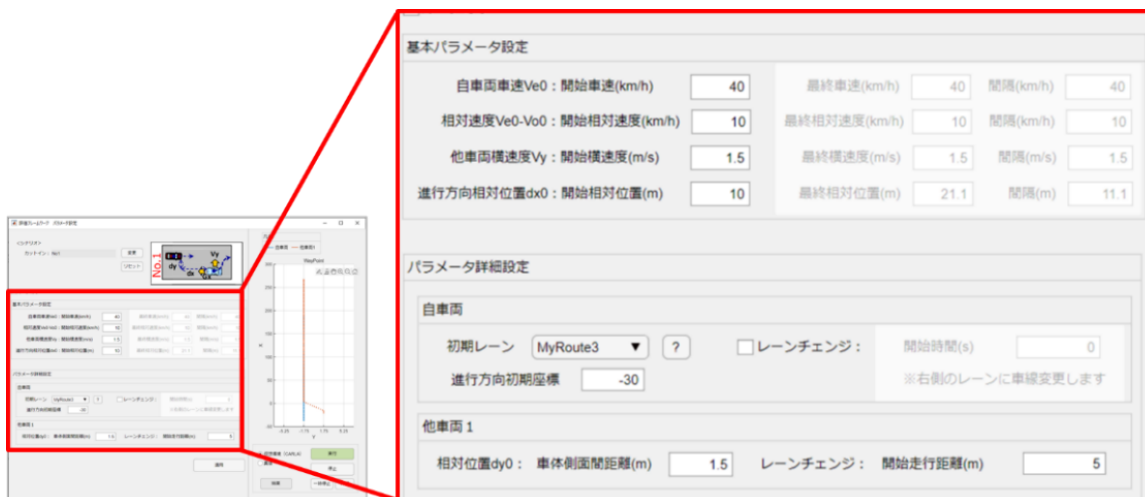
※一般社団法人 日本自動車工業会が公開している「自動運転の安全性評価フレームワーク Ver1.0」より抜粋

[自動設定パラメータ]

- ・他車両速度 V_{o0} ：自車両速度-相対速度で算出します。
- ・他車両レーンチェンジ先：自車両の初期レーンが指定されます。

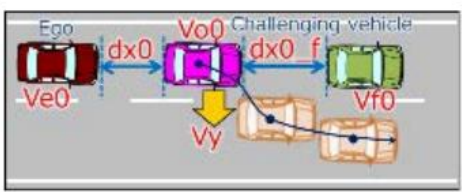
[変更可能パラメータ]

- ・自車両車速 V_{e0} (単位 km/h)
- ・相対速度 $V_{e0}-V_{o0}$ (単位 km/h)
 - ※相対速度は自車両速度より小さい値を指定してください。他車両 1 の車速は自車両速度-相対速度で算出され、他車両 1 の車速は 0km/h 以下は指定できないためです。
- ・他車両 1 横方向速度 V_y (単位 m/s)
 - ※他車両横方向速度は他車両速度より大きく設定することはできません。
 - 単位に注意して指定してください(1km/h = 3.6m/s)。
 - ※横方向速度の上限は 1.8m/s です。¹⁾
- ・進行方向相対位置 dx_0 (単位 m)
- ・自車両初期レーン
- ・進行方向初期座標 (X 座標)
 - ※指定する座標は自車両初期レーンの進行方向の道路範囲を超えることはできません。
 - ※道路終端付近を初期座標とすると、WayPoint がうまく生成されない場合があります。
- ・自車両のレーンチェンジ有無
 - ※チェックを入れると右側レーンに車線変更を行います。左側レーンには車線変更できません。
- ・自車両のレーンチェンジ開始指示時間 (単位 s)
 - ※レーンチェンジありの場合のみ設定できます。
 - ※シミュレーション開始からの経過時間を指定してください。
- ・他車両 1 の車体側面間相対位置 dy_0 (単位 m)
 - ※ +1.5m を指定すると、自車両の初期レーンに対して右側レーン中心線上に他車両 1 が配置されます。
 - 同様に -1.5m を指定すると、自車両の初期レーンに対して左側レーン中心線上に他車両 1 が配置されます。
 - ※設定値は、自車が走行するレーンに対して左右のレーンの範囲内で設定してください。
 - また、MyRoute1(分流路)では右側レーンが、MyRoute4(合流路)では左側レーンが存在しませんので、存在しないレーンの位置には設定しないでください。
 - (右側レーン： 0.75m ~ 2.25m、左側レーン： -0.75m ~ -2.25m)
- ・レーンチェンジ開始走行距離 (単位 m)
 - ※基本 5m 以上を設定してください。
 - 他車両がシミュレーション開始時からレーンチェンジの方向に向いてしまうことを防ぎます。



5.2.2.4 カットアウトシナリオのパラメーター一覧

- ※No2 シナリオのみ他車両が 2 台表示されます。このとき、先行車が他車両 1、先々行車が他車両 2 です。No2 シナリオは、他のカットアウトシナリオ（No6、No13、No20）と設定できるパラメータが若干異なります。



Initial condition	Initial velocity	[Ve0]	Ego vehicle velocity
		[Vo0]	Leading vehicle velocity
		[Vfo]	Vehicle in front of leading vehicle
	Initial distance	[dx0]	Longitudinal distance [※]
[dx0_f]		Front of lead distance	
Vehicle motion	Lateral motion	[Vy]	Lateral velocity

※ Follow the leading vehicle in THW=2sec
 Vo0 = Ve0 (Same speed as the leading vehicle)
 Vfo = 0 (stop vehicle)

※一般社団法人 日本自動車工業会が公開している「自動運転の安全性評価フレームワーク Ver1.0」より抜粋

●他車両 2 台シナリオ (No2)

[自動設定パラメータ]

- ・他車両 1 [先行車] 車速 Vo0 (単位 km/h) : 自車両車速と同じです。
- ・自車両と他車両 1 [先行車] の進行方向相対位置 dx0 (単位 m) : THW=2sec²⁾ となるよう設定されます
- ・他車両 2 [先々行車] 車速 Vfo (単位 km/h) : 0km/h

[変更可能パラメータ]

- ・自車両車速 Ve0 (単位 km/h)
- ・他車両 1 横方向速度 Vy (単位 m/s)

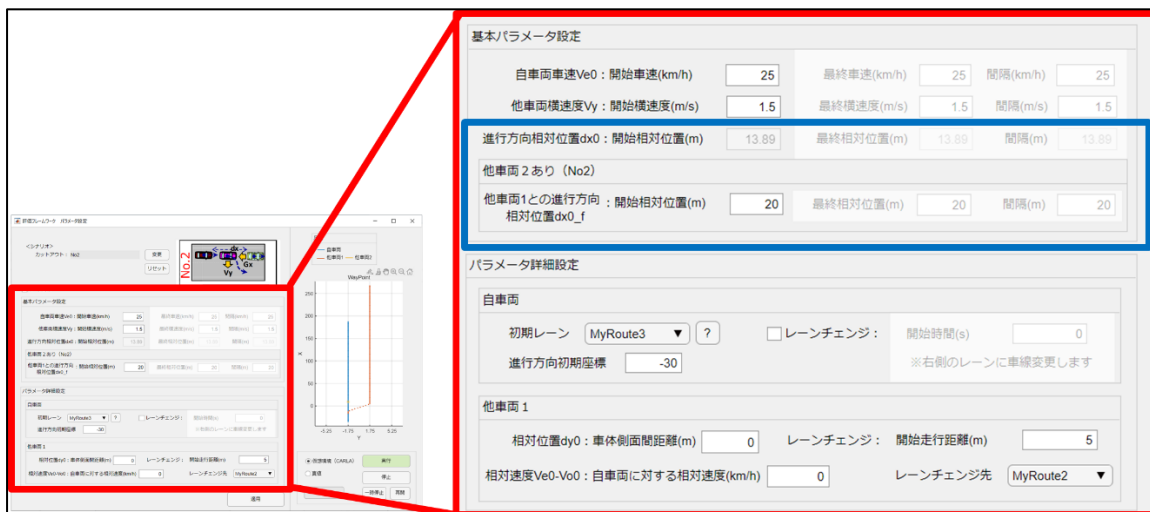
※他車両横方向速度は他車両車速より大きく設定できません。他車両車速は自車両車速と同じです。

単位に注意して指定してください(1km/h = 3.6m/s)。

- ※横方向速度の上限は 1.8m/s です。¹⁾
 - ・他車両 1 [先行車]と他車両 2 [先々行車]の進行方向相対位置 dx0_f (単位 m)
 - ・自車両初期レーン
 - ・進行方向初期座標 (X 座標)
 - ※指定する座標は自車両初期レーンの進行方向の道路範囲を超えることはできません。
 - ※道路終端付近を初期座標とすると、WayPoint がうまく生成されない場合があります。
 - ・自車両のレーンチェンジ有無
 - ※チェックを入れると右側レーンに車線変更を行います。左側レーンには車線変更できません。
 - ・自車両のレーンチェンジ開始指示時間 (単位 s)
 - ※レーンチェンジありの場合のみ設定できます。
 - ※シミュレーション開始からの経過時間を指定してください。
 - ・他車両 1 の車体側面間相対位置 dy0 (単位 m)
 - ※0m を指定すると、自車両の初期レーンの中心線上に他車両 1 が配置されます。
 - ※設定値は、走行するレーンの範囲内で設定してください(-0.75m ~ 0.75m)。
 - ・自車両と他車両 1 [先行車]の相対速度 $V_{e0}-V_{o0}$ (単位 km/h) : 0km/h
 - ・レーンチェンジ開始走行距離 (単位 m)
 - ※基本 5m 以上を設定してください。
 - 他車両がシミュレーション開始時からレーンチェンジの方向に向いてしまうことを防ぎます。
 - ・他車両 1 レーンチェンジ先
- 他車両 1 台シナリオ (No6、No13、No20)
- [自動設定パラメータ]
- ・他車両 1 車速 V_{o0} (単位 km/h) : 自車両車速と同じです。
- [変更可能パラメータ]
- ・自車両車速 V_{e0} (単位 km/h)
 - ・他車両横方向速度 V_y (単位 m/s)
 - ※他車両横方向速度は他車両車速より大きく設定できません。他車両車速は自車両車速と同じです。
 - 単位に注意して指定してください(1km/h = 3.6m/s)。
 - ※横方向速度の上限は 1.8m/s です。¹⁾
 - ・自車両と他車両 1 の進行方向相対位置 dx0
 - ・自車両初期レーン
 - ・進行方向初期座標 (X 座標)
 - ※指定する座標は自車両初期レーンの進行方向の道路範囲を超えることはできません。
 - ※道路終端付近を初期座標とすると、WayPoint がうまく生成されない場合があります。
 - ・自車両のレーンチェンジ有無
 - ※チェックを入れると右側レーンに車線変更を行います。左側レーンには車線変更できません。
 - ・自車両のレーンチェンジ開始指示時間 (単位 s)

- ※レーンチェンジありの場合のみ設定できます。
- ※シミュレーション開始からの経過時間を指定してください。
- ・他車両 1 の車体側面間相対位置 dy_0 (単位 m)
 - ※0mを指定すると、自車両の初期レーンの中心線上に他車両 1 が配置されます。
 - ※設定値は、走行するレーンの範囲内で設定してください(-0.75m ~ 0.75m)。
- ・相対速度 Ve_0-Vo_0 (単位 km/h) : 0km/h
- ・レーンチェンジ開始走行距離 (単位 m)
 - ※基本 5m 以上を設定してください。
 - 他車両がシミュレーション開始時からレーンチェンジの方向に向いてしまうことを防ぎます。
- ・他車両 1 レーンチェンジ先

※青枠内のパラメータについては、他車両の総数によって設定可否が変わります（図は他車両 2 台の場合）。



5.2.2.5 減速シナリオのパラメータ一覧

	Initial condition	Initial velocity	[Ve0] Ego vehicle velocity
			[Vo0] Leading vehicle velocity
	Initial distance		[dx0] Longitudinal distance ^{※1}
Vehicle motion	Deceleration		[Gx_max] Maximum deceleration G
			[dG/dt] Deceleration rate ^{※2}
<p>※1 Follow the leading vehicle in THW=2sec $Vo_0 = Ve_0$ (Same speed as the leading vehicle)</p> <p>※2 The most severe conditions ∞</p>			

※一般社団法人 日本自動車工業会が公開している「自動運転の安全性評価フレームワーク Ver1.0」より抜粋

[自動設定パラメータ]

- ・相対速度 $V_{e0}-V_{o0}$ (単位 km/h) : 0km/h
- ・他車両 1 車速 V_{o0} (単位 km/h) : 自車両車速と同じです。
- ・自車両と他車両の進行方向相対位置 dx_0 (単位 m) : $THW=2\text{sec}^2$ となるよう設定されます。
- ・他車両 1 レーンチェンジ : なし

[変更可能パラメータ]

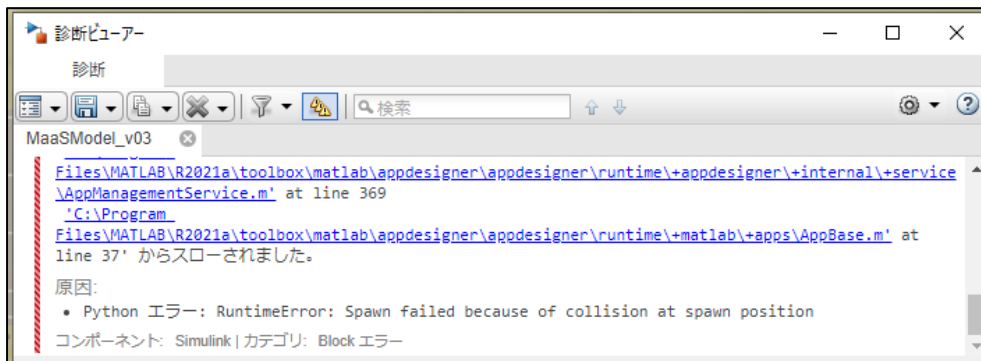
- ・自車両車速 V_{e0} (単位 km/h)
- ・他車両 1 最大加減速度 G_{x_max} (単位 G)
※加減速度はゼロより小さい値を設定してください。
- ・他車両 1 横方向速度 V_y (単位 m/s)
※他車両横方向速度は他車両速度より大きく設定することはできません。
単位に注意して指定してください(1km/h = 3.6m/s)。
※横方向速度の上限は 1.8m/s です。¹⁾
- ・自車両初期レーン
- ・進行方向初期座標 (X 座標)
※指定する座標は自車両初期レーンの進行方向の道路範囲を超えることはできません。
※道路終端付近を初期座標とすると、WayPoint がうまく生成されない場合があります。
- ・自車両のレーンチェンジ有無
※チェックを入れると右側レーンに車線変更を行います。左側レーンには車線変更できません。
- ・自車両のレーンチェンジ開始指示時間 (単位 s)
※レーンチェンジありの場合のみ設定できます。
※シミュレーション開始からの経過時間を指定してください。
- ・他車両 1 の車体側面間相対位置 dy_0 (単位 m)
※0m を指定すると、自車両の初期レーンの中心線上に他車両 1 が配置されます。
※設定値は、走行するレーンの範囲内で設定してください(-0.75m ~ 0.75m)。



◇ シミュレーション実行時の注意点

※必ず「適用」ボタンを押してから実行してください。

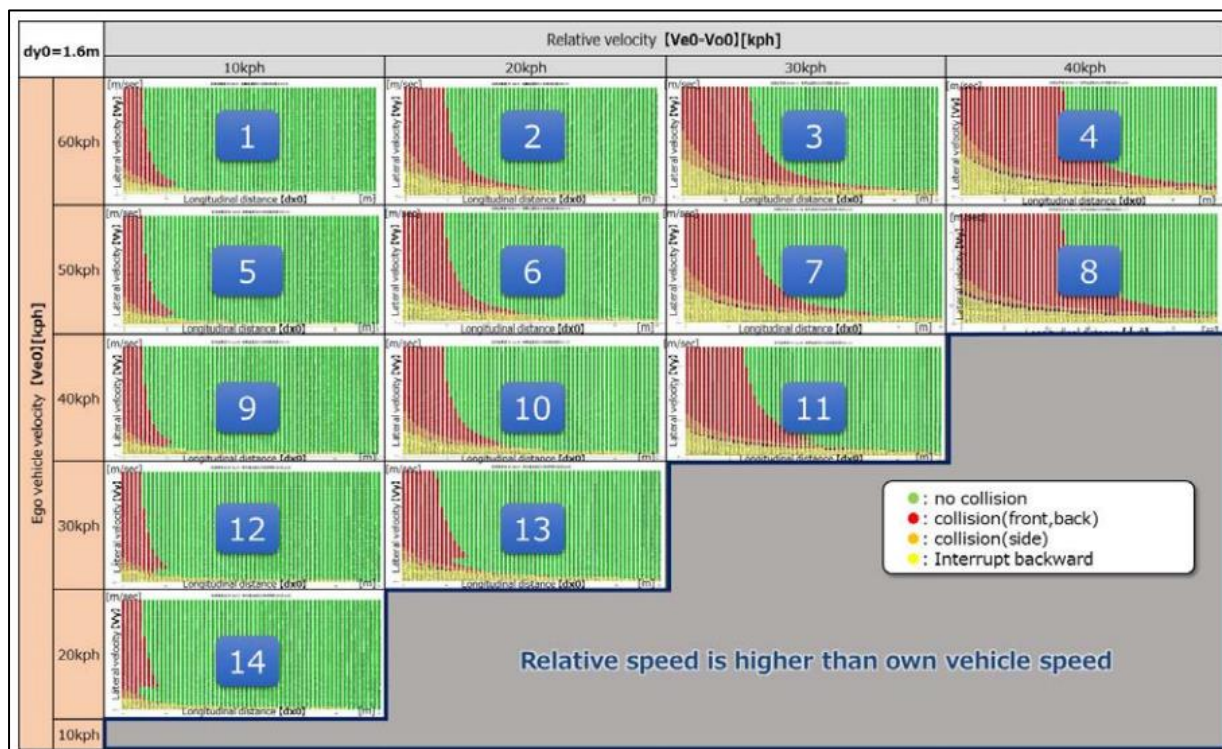
※仮想環境で実行した際に以下のエラーが起こった場合、自車両か他車両の初期位置にて CARLA 上の他オブジェクトと接触している場合があります。（例：道路脇のポールと初期位置がかぶっている）その場合、進行方向相対位置 dx0 または 車体側面相対位置 dy0 を修正してください。



5.2.2.6 ループ検証

自動運転の安全性を評価するためには、安全性に関わるパラメータを徐々に変えながら衝突の有無を繰り返し検証していく必要があります。このような場合に、パラメータを変動させる範囲と粒度を指定してループ検証を行うことができます。変動可能なパラメータは、「自動運転の安全性評価フレームワーク Ver1.0」で提示されている項目を採用しています。

- カットインシナリオ

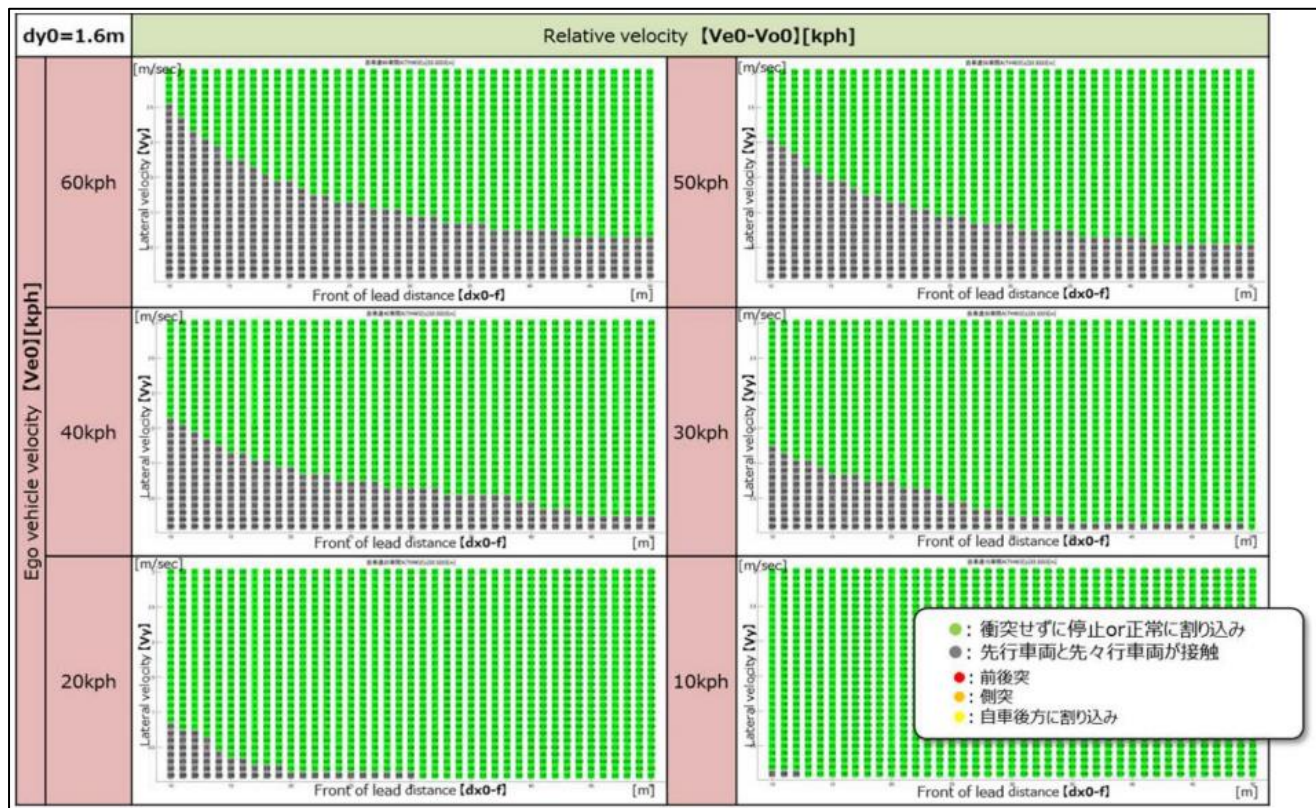


※一般社団法人 日本自動車工業会が公開している「自動運転の安全性評価フレームワーク Ver1.0」より抜粋

[ループ検証用パラメータ]

- 自車両速度 V_{e0} (単位 km/h)
- 相対速度 $V_{e0}-V_{o0}$ (単位 km/h)
 - ※相対速度が自車両車速以上となるパターンが指定された場合、その検証はスキップされます。
- 他車両横方向速度 V_y (単位 m/s)
 - ※他車両横方向速度は他車両車速より大きく設定することはできません。
 - そのため全パターンのうち、他車両横方向速度が他車両車速より大きくなるパターンが生じる場合、「適用」ボタンを押したときにエラーが表示され、パラメータ設定は適用されません。
 - その場合、他車両横方向速度または相対速度を再検討してください。

- カットアウトシナリオ

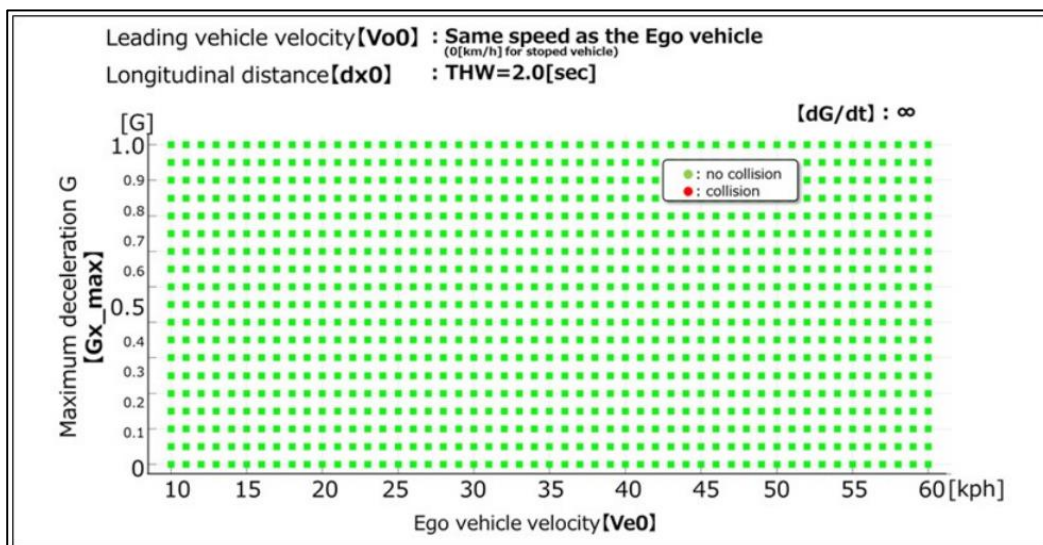


※一般社団法人 日本自動車工業会が公開している「自動運転の安全性評価フレームワーク Ver1.0」より抜粋

[ループ検証用パラメータ]

- ・自車両速度 V_{e0} (単位 km/h)
- ・先行車との相対速度 $V_{e0}-V_{o0}$ (単位 km/h)
- ※相対速度が自車両車速以上となるパターンが指定された場合、その検証はスキップされます。
- ・他車両横方向速度 V_y (単位 m/s)
- ※他車両横方向速度は他車両速度より大きく設定することはできません。全パターンのうち他車両横方向速度が他車両速度より大きくなるパターンが生じる場合、「適用」ボタンを押したときにエラーが表示され、パラメータ設定は適用されません。その場合、他車両横方向速度または相対速度を再検討してください。
- ・先行車との相対位置 dx_0
- ※No2 シナリオの場合は先行車両(他車両 1)と先々行停車車両(他車両 2)との相対位置 dx_0-f

- 減速シナリオ



※一般社団法人 日本自動車工業会が公開している「自動運転の安全性評価フレームワーク Ver1.0」より抜粋

[ループ検証用パラメータ]

- ・自車両速度 Ve0 (単位 km/h)
- ・加減速度 Gx_max (単位 G)

- ループ検証実行方法

パラメータ設定ウィンドウの「ループあり」にチェックを入れると、検証の範囲と粒度を指定できるようになります。



[検証方法例] 減速シナリオ No4

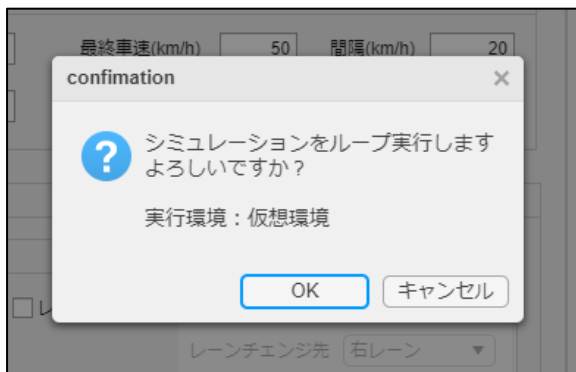
下図のように設定した場合

ループ検証パラメータ設定					
自車両車速 V_{e0} : 開始車速(km/h)	<input type="text" value="10"/>	最終車速(km/h)	<input type="text" value="60"/>	間隔(km/h)	<input type="text" value="20"/>
他車両1最大加減速度 G_{x_max} : 開始加減速度(G)	<input type="text" value="-0.5"/>	最終加減速度(G)	<input type="text" value="-2"/>	間隔(G)	<input type="text" value="-0.5"/>

- ・自車両車速検証パターン [10, 30, 50]
 - ・加減速度検証パターン [-0.5, -1, -1.5, -2]
- 検証ループ総数 12

パラメータを変更したら、「適用」ボタンを押してください。指定したパラメータに問題があると、エラー表示が出て適用が中止されます。

適用が完了し、「実行」ボタンを押すと、以下のようにループで実行することの確認ウィンドウが表示されます。「OK」を押すとシミュレーションが開始されます。

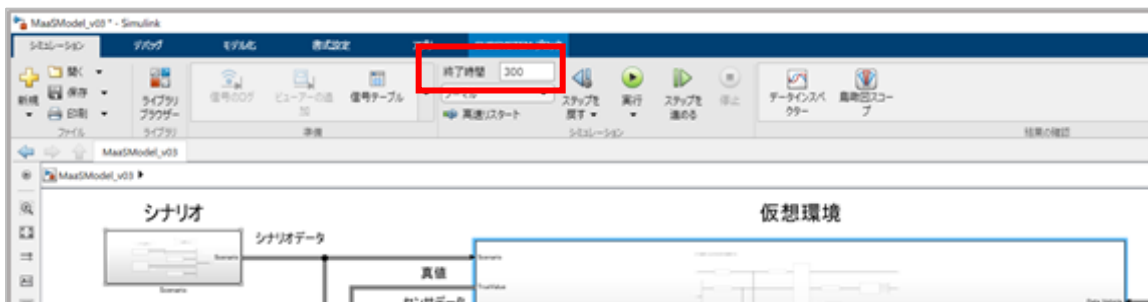


※ループ検証でも実行環境（仮想環境/真値）を選択できます。

※シミュレーション中に「停止」ボタンを押すと、現在実行中のシミュレーションは停止し、次のパターンのシミュレーションに移行します。

※ループ検証をすべて強制終了したい場合は、パラメータ設定ウィンドウを閉じてください。

※1回の検証時間が長すぎる場合は、Simulinkモデルの「シミュレーション」タブ内にある終了時間を短く設定してください（デフォルトでは300sです）。短く設定しすぎると、正確な結果が得られる前にシミュレーションが終了してしまう可能性がありますので、ご注意ください。



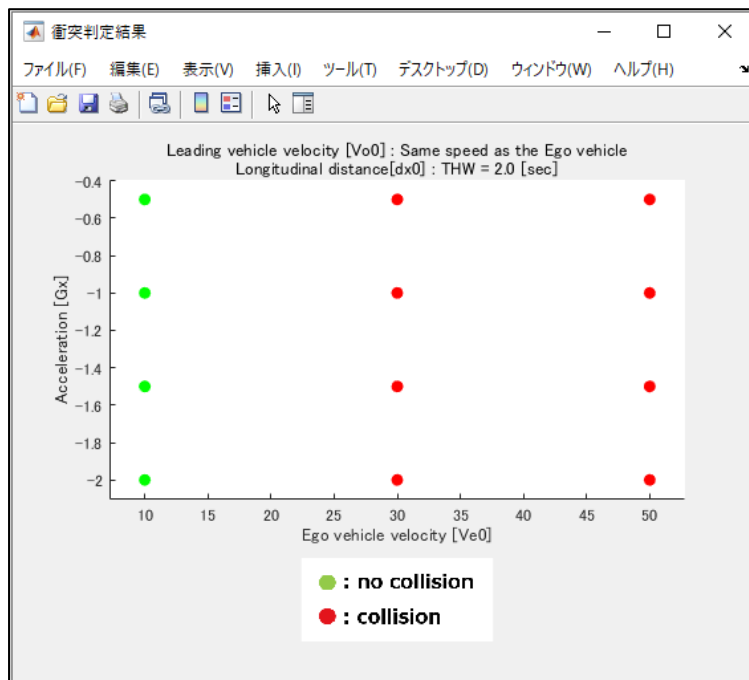
- 結果の確認

ループ検証では、各パターンのシミュレーションが終わるごとに衝突判定の結果のみが[衝突判定結果]ウィンドウに記録されていきます。

※ループ検証では「結果」ボタンから結果を確認することはできません。

※ループ検証中は[衝突判定結果]ウィンドウを削除しないでください。記録ができずループ検証が停止します。

[結果表示例]減速シナリオ No4



【補足情報】

1) 「自動運転の安全性評価フレームワーク Ver1.0」 p.11

図 15. 「最大横速度」の実観測データ統計値 を参考に上限を設定しました。

2) THW は、自車両車速と先行車速が等しい時、車間距離を走行速度で割った数値です。

先行車の現在位置に自車両が到達するまでの時間を示します。

6 FAQ(トラブルシューティング)

Q1.Python が実行できません。

A1.インストールが正しく行われているか確認してください。

セットアップ時に、Add Python 3.7 to PATH にチェックを入れずにインストールした場合は、システム環境変数[Path]に、Python3.7 の実行ファイルのパスを追加してください。

Q2.pip3 コマンドを実行する際、Retry エラーが出ます。

A2.インターネット接続ができない可能性があります。ネットワーク化プロキシの設定を確認してください。

Q3.シミュレーション停止後に、再度シミュレーションを実行するとエラーが出ます。

A3.処理タイミングによるものと思われるので、再度シミュレーションを実行してください。

それでもエラーが出る場合は、すべてのアプリケーションを閉じたのち、再度実行を試してください。

Q4.シミュレーション実行時に Downloading https://ultralytics.com/assets/Arial.ttf to C:¥xx¥Arial.ttf..

エラー: request>do_open (行 1352)

Python エラー: URLError: <urlopen error [Errno 11001] getaddrinfo failed>

のエラーが出てシミュレーションが実行できません。

A4.本解説書の P37 を参照ください。

Q5.CARLA 実行時に、VisualC++ Redistributable for Visual Studio V2019 のインストールを
求めるメッセージが出ます。

A5.本解説書の P35 を参照ください。

Q6.CARLR をビルド中に、kernel32.lib が見つかりませんの表示でビルドが中断します。

A6.コマンドプロンプトを、VisualStudio2019 のものを使用してください。

スタートメニューの[VisualStudio2019]内、x86_x64CrossToolCommandPrompt を使用してください。

エラーによる再トライの場合は、C:¥carla¥carla¥Build 内のファイルを削除してください。

Q7. CARLA をビルドしてマップをインポートした場合に、CRANTS マップの信号機が表示されません。

A7. C:\%carla%\carla\Unreal\CarlaUE4\Content\Carla 内の「ExportedMaps」ファイルを削除してください。

Q8. シミュレーションを実行した後に Simulink モデルを閉じようとしたところ、
モデルを変更していないのに「閉じる前に保存しますか？」の注意書きが出ます。

A8. モデルのパラメータを変更しているためです。保存せず「いいえ」で閉じてください。「はい」で保存しても、問題はありません。次回以降にシミュレーション実行時にエラーが出る場合は、パラメータ設定画面で一度「適用」ボタンを押してから実行してください。

Q9. CARLA を立ち上げているのに、CARLA に接続できないというエラーが出ます。

A9. タスクマネージャーを確認してください。「プロセス」内に UE4Editor が 2 つ存在していた場合、CARLA が 2 重に開いている状態となり、エラーの原因となります。現在起動されているすべての「UE4Editor」を、右クリック→タスクの終了で閉じたのち、再度 CARLA を起動しなおしてください。

Q10. シミュレーション実行中に CARLA を閉じたら、CARLA が動作できなくなりました。

A10. タスクマネージャーを確認してください。「UE4Editor」が「応答なし」となっている場合、右クリック→タスクの終了で閉じたのち、再度 CARLA を起動しなおしてください。

Q11. シナリオを読み込み時、シミュレーションを実行時などに、「ファイルが見つかりません」とエラーが出ます。

A11. MATLAB の[現在のフォルダー]が、Simulink モデル(METI_AutonomousDrivingSimulationModel.slx)がある階層になっていることを確認してください。それでもエラーが出る場合は init_setting.m を再度実行しなおしてください。

Q12. モデル実行時に、「GUIHandle がない」とエラーが出ます。

A12. init_setting.m を再度実行しなおしてください。

Q13. ループ検証をしていて、パラメータ設定画面を閉じてもループが止まらなくなっていました。

A13. MATLAB のコマンドウィンドウを選択し、Ctrl+C を押すことでループ実行を強制終了させることができます。

Q14. CARLA を起動しようとしたら、「Out of video memory」のエラーが出ます。

A14. CARLA の起動に必要なビデオメモリの確保ができていない状態です。この場合、お使いの動作環境では CARLA は利用できません。シミュレーションは真値で実行してください。

Q15. シナリオのループ実行など、長時間のシミュレーションを行うと、CARLA が強制終了してしまいます。

A15. メモリ不足など、システムなどの要因でまれに発生する場合があります。短い時間でのシミュレーションでの利用もしくは、真値を利用したモードでのシミュレーションをご利用ください。

Q16. MATLAB/Simulink 2021b での使用は可能ですか？

A16. 申し訳ございません。2021a 以外での動作確認は行っておりません。

Q17. なぜ Mathworks RoadRunner だけ 2021b なのですか？

A17. 日本の道路標識 Asset が追加されたため、RoadRunner のみ 2021b を使用しています。

Q18. 実行時に、No such file or directory: 'virtual¥¥yolov5¥¥weights.pt'のエラーがでます。

A18. MATLAB のカレントディレクトリをモデルのルートになっているか確認してください。

Q19. 実行時に、「Upsample」 object has no attribute 'recompute_scale_factor」のエラーが出ます。

A19. GPU の最新のドライバをインストールして、再度実行してください。

7 APPENDIX

7.1 参考文献

- 1) 一般社団法人 日本自動車工業会 「自動運転の安全性評価フレームワーク Ver1.0」

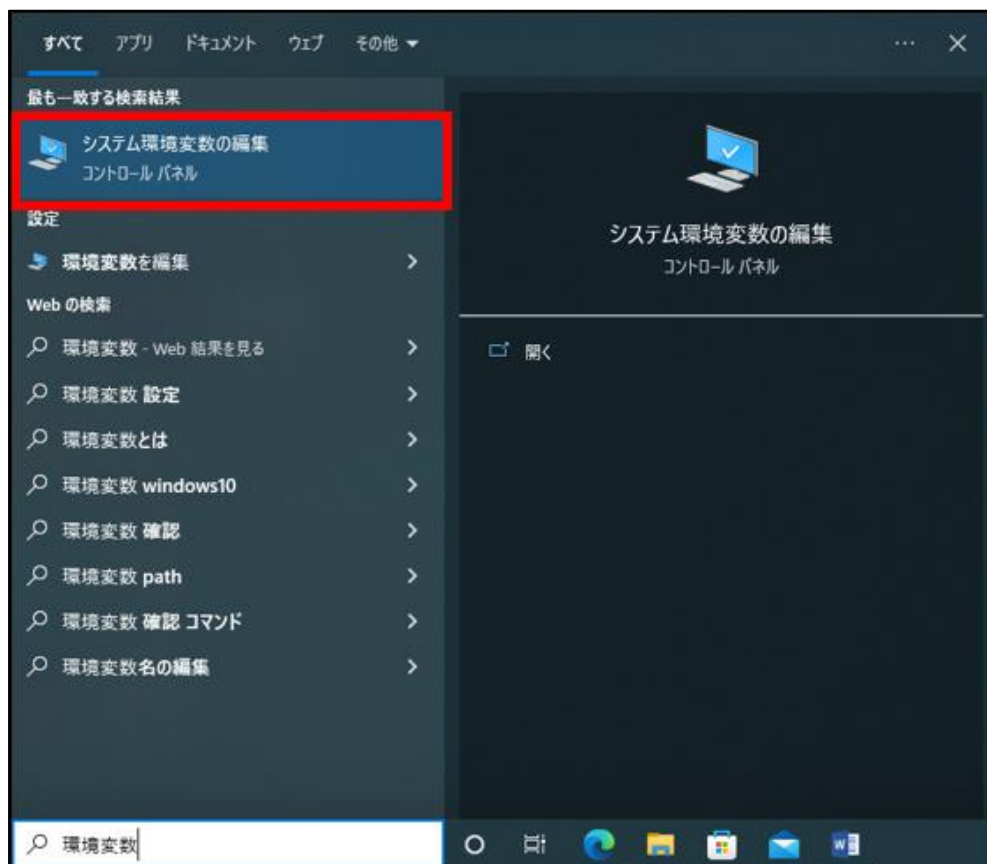
7.2 複数 PYTHON インストール時の手順

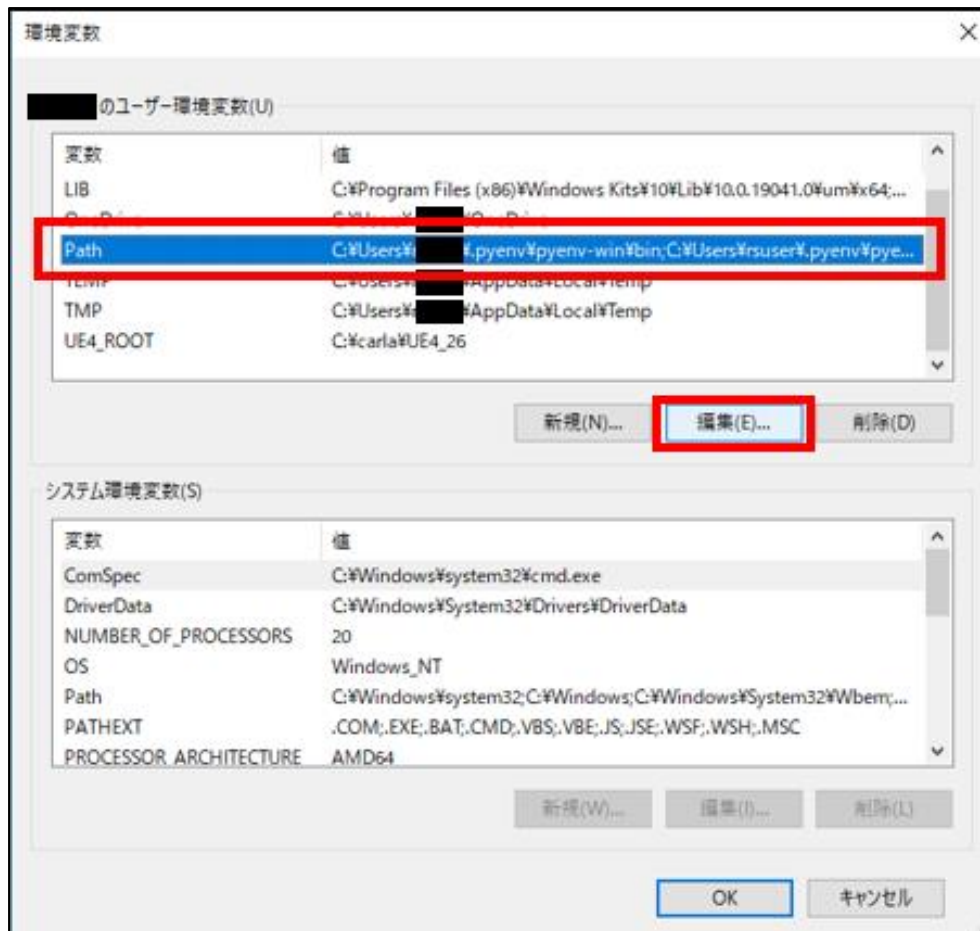
- 1) Python のバージョンを変更する

①,②いずれかの方法で Python バージョンの変更を行ってください。なお、②を行う場合は事前に pyenv をインストールしてください。

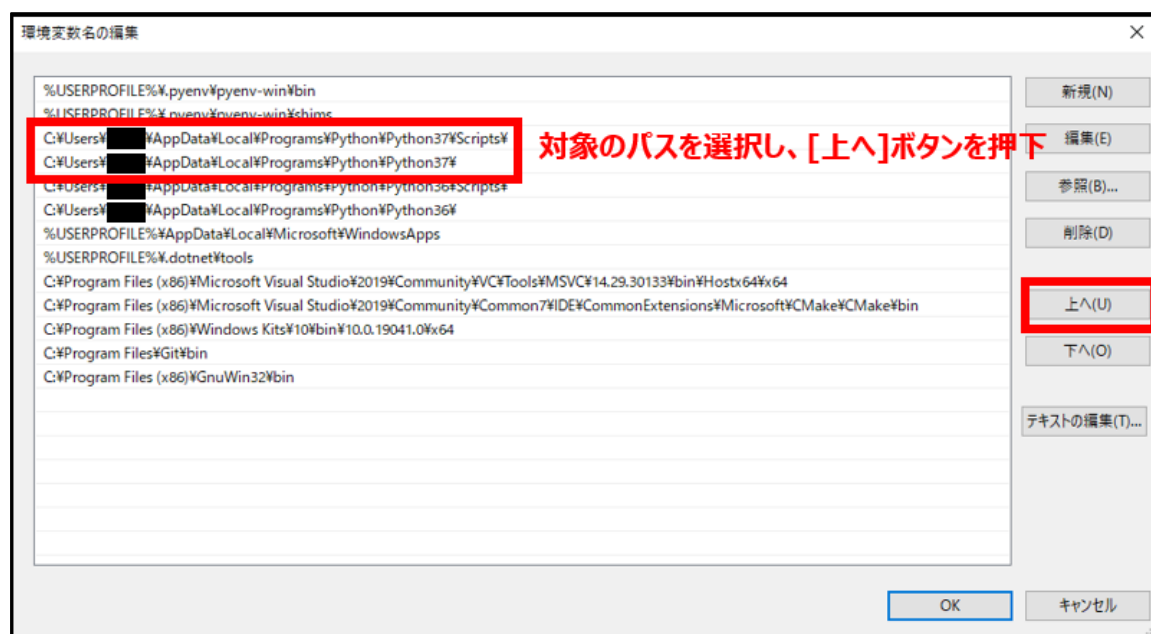
- ① PC の環境設定から python のバージョンを変更

・[システム環境変数の編集]を開き、ユーザー環境変数の[Path]を選択した状態で[編集]をクリックしてください。





- ・“C:¥Users¥<ユーザー名>¥AppData¥Local¥Programs¥Python¥Python37¥Scripts¥”と
“C:¥Users¥<ユーザー名>¥AppData¥Local¥Programs¥Python¥Python37¥”を、他バージョン
の Python パスよりも上へ移動してください。



② pyenv コマンドにより python のバージョンを変更

・コマンドプロンプトを開いてください。

pyenv による Python3.7.9 のインストールを行っていない場合は、以下のコマンドを実行してください。

```
>pyenv install 3.7.9
```

・以下のコマンドを実行してください。

```
>pyenv global 3.7.9
```

・MATLAB を開き、コマンドウィンドウで以下のコマンドを実行してください。

```
>pyenv('Version','3.7')
```

2) バッチファイルの修正

c:\¥carla¥carla¥Util¥BuildTools 内の

・BuildPythonAPI.bat 112 行目の"py -3"を"py -3.7"に変更してください。

・BuildCarlaUE4.bat 113 行目と 116 行目の"py -3"を"py -3.7"に変更してください。

・Package.bat 116 行目と 119 行目の"py -3"を"py -3.7"に変更してください。

7.3 PYTHON3.7 選定理由

本自動運転制御モデルは、MATLAB2021a・CARLA・yolov5 の対応を考慮し Python3.7 を選定しました。MATLAB2021a、CARLA、yolov5 がサポートしている Python のバージョンはそれぞれ以下の通りです。

MATLAB2021a : Python2.7 , 3.7 , 3.8

CARLA : Python3

yolov5 : Python3.7 以上