

artisoc Cloudレシピブック

07. 出力設定をマスターしよう

（株）構造計画研究所
創造工学部
https://mas.kke.co.jp

Copyright© KOZO KEIKAKU ENGINEERING Inc.All Rights Reserved.
この資料には、弊社のノウハウ、営業機密等が含まれておりますので、お取り扱いには十分ご留意願います。この資料およびその内容を、弊社に無断で使用、複写、破壊、改ざんすること、ならびに第三者へ開示すること、漏洩すること、あるいは使用させることは、固くお断り申しあげます。

artisoc Cloudがサポートする出力形式

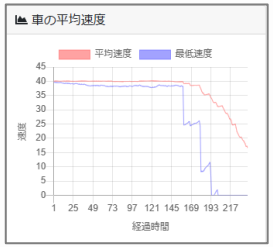
構造計画研究所
KOZO KEIKAKU ENGINEERING Inc.

artisoc Cloudは、以下の出力形式をサポートしています

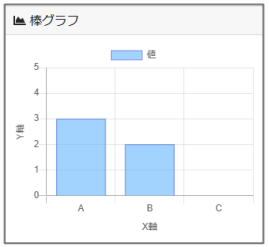
No.	出力形式
1	マップ出力
2	時系列グラフ
3	棒グラフ
4	積み上げ棒グラフ
5	横棒グラフ
6	積み上げ横棒グラフ
7	円グラフ
8	散布図
9	折れ線グラフ
10	積み上げ折れ線グラフ
11	ヒストグラム
12	値出力



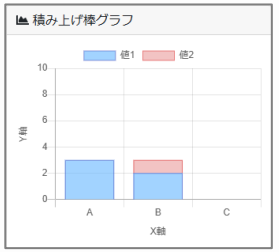
1. マップ出力



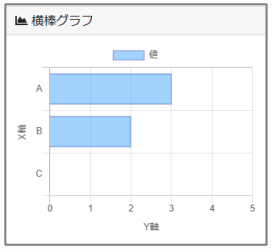
2. 時系列グラフ



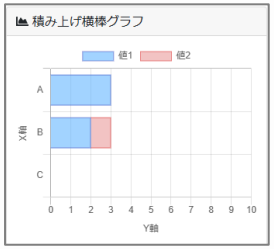
3. 棒グラフ



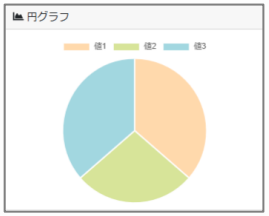
4. 積み上げ棒グラフ



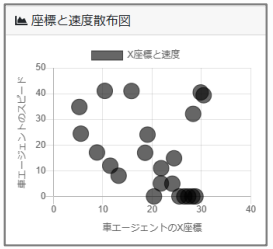
5. 横棒グラフ



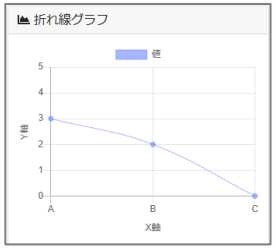
6. 積み上げ横棒グラフ



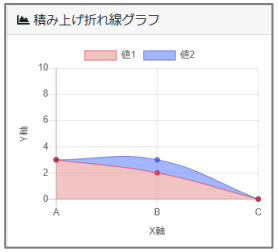
7. 円グラフ



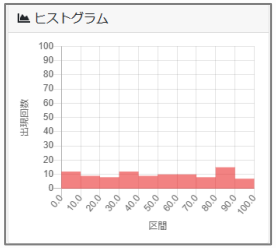
8. 散布図



9. 折れ線グラフ



10. 積み上げ折れ線グラフ



11. ヒストグラム



12. 値出力

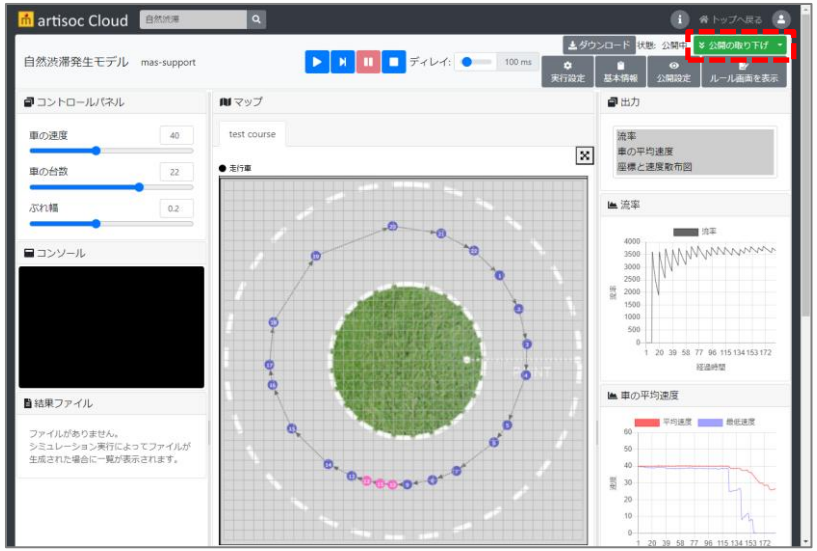
Copyright© KOZOKEIKAKU ENGINEERING Inc. All Rights Reserved.

自然渋滞発生モデルに出力設定を追加

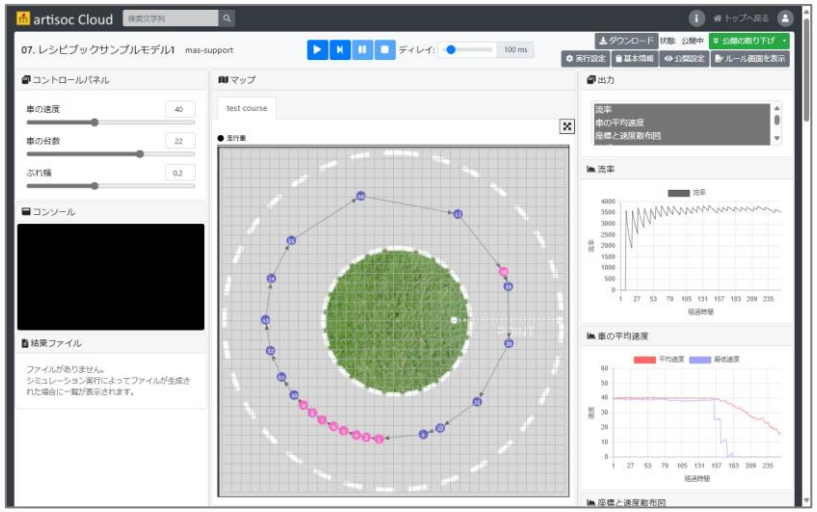
構造計画研究所
KOZO KEIKAKU ENGINEERING Inc.

自然渋滞発生モデルをベースに出力設定を追加します

- 画面右上の「継承して新規作成」をクリックして新しいモデルを作成する



自然渋滞発生モデル



07. レシピブック　サンプルモデル1

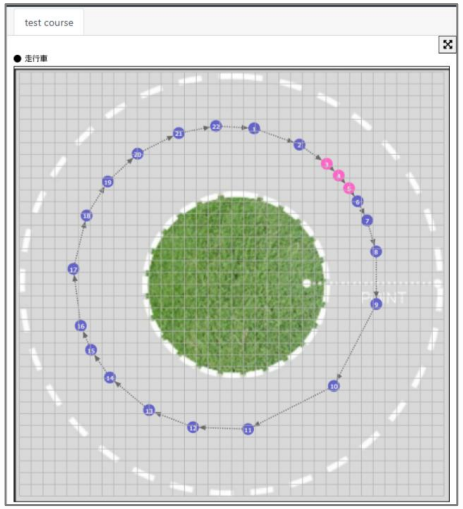
Copyright© KOZOKEIKAKU ENGINEERING Inc. All Rights Reserved.

1. マップ出力

マップ出力は定義されているので設定を確認します

① 出力パネルの右上にマウスカーソルを移動し、設定アイコンをクリックする
② 出力設定画面のマップ出力設定項目リストから test course の 編集アイコンをクリックする
③ マップ出力設定画面のマップ要素リストの 走行車 の編集アイコンをクリックする
④ マップ要素設定画面のエージェントに car が指定されていることを確認する

※ 拡大率、透明度、情報の表示、線を引くなどを変更すると見た目が変わります



マップ出力



① 出力パネル　② 出力設定画面　③ マップ出力設定画面　④ マップ要素設定画面

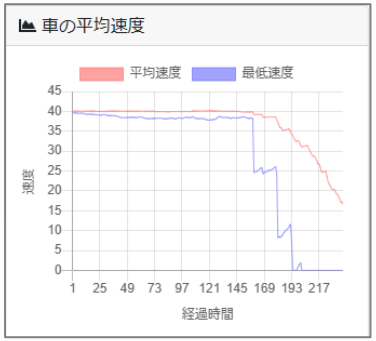
Copyright© KOZOKEIKAKU ENGINEERING Inc. All Rights Reserved.

2. 時系列グラフ

構造計画研究所
KOZO KEIKAKU ENGINEERING Inc.

時系列グラフは定義されているので設定を確認します

① 出力パネルの右上にマウスカーソルを移動し、設定アイコンをクリックする
② 出力設定画面のグラフ出力設定項目リストから 車の平均速度 の編集アイコンをクリックする
③ 時系列グラフ設定画面の時系列グラフ要素リストから 平均速度 (または 最低速度)の編集アイコンをクリックする
④ 時系列要素設定画面の出力値に Universe.ave_car_v (最低速度の場合は Universe.min_car_v)が指定されていることを確認する



時系列グラフ

※ 時系列グラフはstep毎にデータが蓄積されるので、長いシミュレーションを行う場合は注意してください



① 出力パネル　② 出力設定画面　③ 時系列グラフ設定画面　④ 時系列要素設定画面

Copyright© KOZOKEIKAKU ENGINEERING Inc. All Rights Reserved.

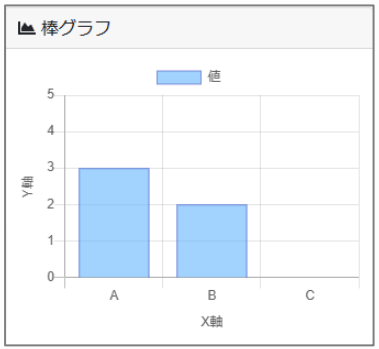
3. 棒グラフ（1）

構造計画研究所
KOZO KEIKAKU ENGINEERING Inc.

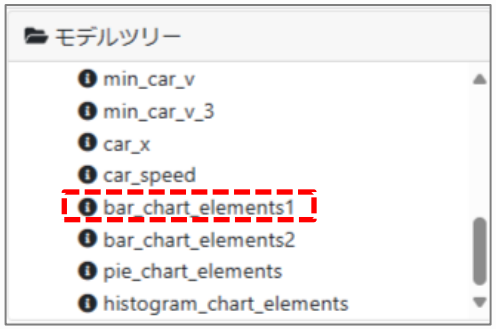
棒グラフを追加します

① モデルツリーの Universe 直下に変数を追加する
- 変数：bar_chart_elements1

② Universe をクリックしてルールエディタを表示し、univ_step_end に下記の通りルールを追加する
- 0～5の整数値をランダムに3つ生成する



棒グラフ



① モデルツリー

```
def univ_step_end(self):
    Universe.ave_car_v = Universe.timechart_element_function0()
    Universe.min_car_v = Universe.timechart_element_function1()

    Universe.car_x = [c.x for c in Universe.jam_space.agents]
    Universe.car_speed = [c.speed for c in Universe.jam_space.agents]

    # レシピブック07.追加
    Universe.bar_chart_elements1 = [random.randint(0, 5) for i in range(3)]
    Universe.bar_chart_elements2 = [random.randint(0, 5) for i in range(3)]
    Universe.pie_chart_elements = [random.randint(0, 5) for i in range(3)]
    Universe.histogram_chart_elements = [random.randint(0, 100) for i in range(100)]
```

② ルールエディタ（Universe）

Copyright© KOZOKEIKAKU ENGINEERING Inc. All Rights Reserved.

3. 棒グラフ（2）

③ 出力パネルの右上にマウスカーソルを移動し、設定アイコン をクリックする
④ 出力設定画面にて、棒グラフ を選択してから 追加ボタン をクリックする
⑤ 棒グラフ設定画面にて、以下の通り設定する

X軸設定　項目ラベル：A,B,C　※ データ数だけカンマ区切りで項目ラベルを定義する

⑥ 棒グラフ要素リストの +アイコン をクリックして、棒グラフ要素設定画面を表示し、以下の通り設定する

出力値：Universe.bar_chart_elements1



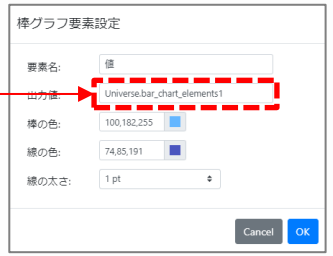
③ 出力パネル



④ 出力設定画面



⑤ 棒グラフ設定画面



⑥ 棒グラフ要素設定画面

Copyright© KOZOKEIKAKU ENGINEERING Inc. All Rights Reserved.

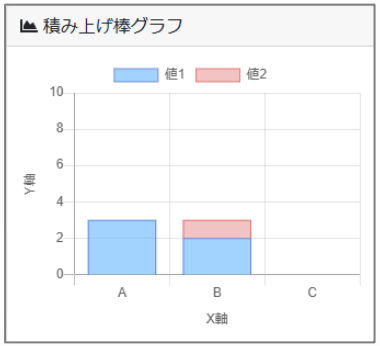
4. 積み上げ棒グラフ（1）

構造計画研究所
KOZO KEIKAKU ENGINEERING Inc.

積み上げ棒グラフを追加します

① モデルツリーの Universe 直下に変数を追加する
- 変数：bar_chart_elements1
- 変数：bar_chart_elements2

② Universe をクリックしてルールエディタを表示し、univ_step_end に下記の通りルールを追加する
- 0～5の整数値をランダムに3つ生成する



積み上げ棒グラフ



① モデルツリー

```
54  def univ_step_end(self):
55      Universe.ave_car_v = Universe.timechart_element_function0()
56      Universe.min_car_v = Universe.timechart_element_function1()
57
58      Universe.car_x = [c.x for c in Universe.jam_space.agents]
59      Universe.car_speed = [c.speed for c in Universe.jam_space.agents]
60
61      # レシピブック07.追加
62      Universe.bar_chart_elements1 = [random.randint(0, 5) for i in range(3)]
63      Universe.bar_chart_elements2 = [random.randint(0, 5) for i in range(3)]
64      Universe.pie_chart_elements = [random.randint(0, 5) for i in range(3)]
65      Universe.histogram_chart_elements = [random.randint(0, 100) for i in range(100)]
66
```

② ルールエディタ（Universe）

Copyright© KOZOKEIKAKU ENGINEERING Inc. All Rights Reserved.

4. 積み上げ棒グラフ（2）

③ 出力パネルの右上にマウスカーソルを移動し、設定アイコン をクリックする
④ 出力設定画面にて、棒グラフ を選択してから 追加ボタン をクリックする
⑤ 棒グラフ設定画面にて、以下の通り設定する
　　積み上げ表示：チェックON
　　X軸設定　項目ラベル：A,B,C　※ データ数だけカンマ区切りで項目ラベルを定義する
⑥ 棒グラフ要素リストの +アイコン をクリックして、棒グラフ要素設定画面を表示し、以下の通り設定する
　　出力値：Universe.bar_chart_elements1
⑦ 同様に、棒グラフ要素設定画面を表示し、以下の通り設定する
　　出力値：Universe.bar_chart_elements2



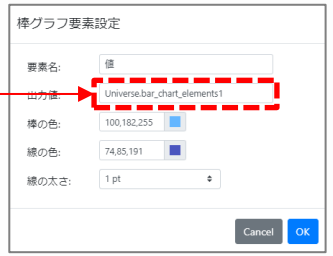
③ 出力パネル



④ 出力設定画面



⑤ 棒グラフ設定画面



⑥ 棒グラフ要素設定画面

Copyright© KOZOKEIKAKU ENGINEERING Inc. All Rights Reserved.

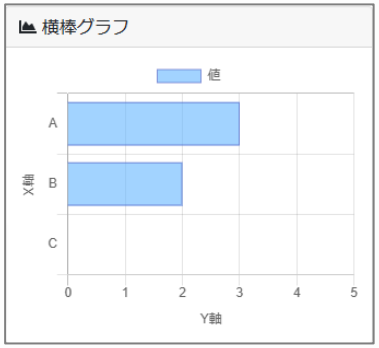
5. 横棒グラフ（1）

構造計画研究所
KOZO KEIKAKU ENGINEERING Inc.

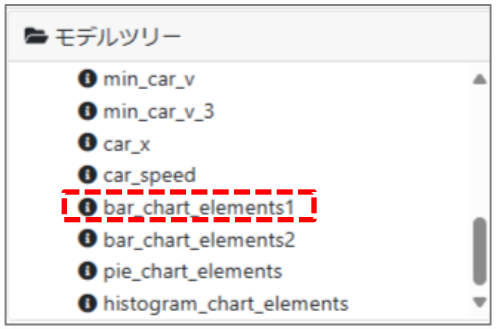
横棒グラフを追加します

① モデルツリーの Universe 直下に変数を追加する
- 変数：bar_chart_elements1

② Universe をクリックしてルールエディタを表示し、univ_step_end に下記の通りルールを追加する
- 0～5の整数値をランダムに3つ生成する



棒グラフ



① モデルツリー

```
def univ_step_end(self):
    Universe.ave_car_v = Universe.timechart_element_function0()
    Universe.min_car_v = Universe.timechart_element_function1()

    Universe.car_x = [c.x for c in Universe.jam_space.agents]
    Universe.car_speed = [c.speed for c in Universe.jam_space.agents]

    # レシピブック07.追加
    Universe.bar_chart_elements1 = [random.randint(0, 5) for i in range(3)]
    Universe.bar_chart_elements2 = [random.randint(0, 5) for i in range(3)]
    Universe.pie_chart_elements = [random.randint(0, 5) for i in range(3)]
    Universe.histogram_chart_elements = [random.randint(0, 100) for i in range(100)]
```

② ルールエディタ（Universe）

Copyright© KOZOKEIKAKU ENGINEERING Inc. All Rights Reserved.

5. 横棒グラフ（2）

③ 出力パネルの右上にマウスカーソルを移動し、設定アイコン をクリックする
④ 出力設定画面にて、棒グラフ を選択してから 追加ボタン をクリックする
⑤ 棒グラフ設定画面にて、以下の通り設定する
　　棒グラフの方向：横棒
　　X軸設定　項目ラベル：A,B,C　※ データ数だけカンマ区切りで項目ラベルを定義する
⑥ 棒グラフ要素リストの +アイコン をクリックして、棒グラフ要素設定画面を表示し、以下の通り設定する
　　出力値：Universe.bar_chart_elements1



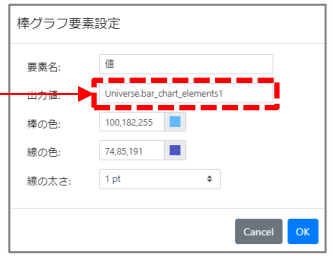
③ 出力パネル



④ 出力設定画面



⑤ 棒グラフ設定画面



⑥ 棒グラフ要素設定画面

Copyright© KOZOKEIKAKU ENGINEERING Inc. All Rights Reserved.

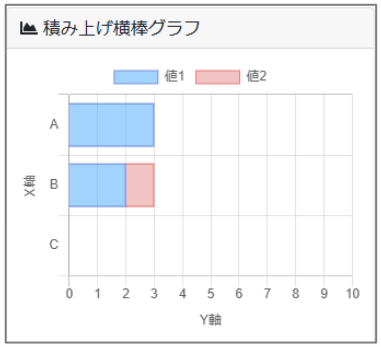
6. 積み上げ横棒グラフ（1）

構造計画研究所
KOZO KEIKAKU ENGINEERING Inc.

積み上げ棒グラフを追加します

① モデルツリーの Universe 直下に変数を追加する
- 変数：bar_chart_elements1
- 変数：bar_chart_elements2

② Universe をクリックしてルールエディタを表示し、univ_step_end に下記の通りルールを追加する
- 0～5の整数値をランダムに3つ生成する



積み上げ横棒グラフ



① モデルツリー

```
54  def univ_step_end(self):
55      Universe.ave_car_v = Universe.timechart_element_function0()
56      Universe.min_car_v = Universe.timechart_element_function1()
57
58      Universe.car_x = [c.x for c in Universe.jam_space.agents]
59      Universe.car_speed = [c.speed for c in Universe.jam_space.agents]
60
61      # レシピブック07.追加
62      Universe.bar_chart_elements1 = [random.randint(0, 5) for i in range(3)]
63      Universe.bar_chart_elements2 = [random.randint(0, 5) for i in range(3)]
64      Universe.pie_chart_elements = [random.randint(0, 5) for i in range(3)]
65      Universe.histogram_chart_elements = [random.randint(0, 100) for i in range(100)]
66
```

② ルールエディタ（Universe）

Copyright© KOZOKEIKAKU ENGINEERING Inc. All Rights Reserved.

6. 積み上げ横棒グラフ（2）

③ 出力パネルの右上にマウスカーソルを移動し、設定アイコン をクリックする
④ 出力設定画面にて、棒グラフ を選択してから 追加ボタン をクリックする
⑤ 棒グラフ設定画面にて、以下の通り設定する
　　積み上げ表示：チェックON
　　棒グラフの方向：横棒
　　X軸設定　項目ラベル：A,B,C　※ データ数だけカンマ区切りで項目ラベルを定義する
⑥ 棒グラフ要素リストの +アイコン をクリックして、棒グラフ要素設定画面を表示し、以下の通り設定する
　　出力値：Universe.bar_chart_elements1
⑦ 同様に、棒グラフ要素設定画面を表示し、以下の通り設定する
　　出力値：Universe.bar_chart_elements2



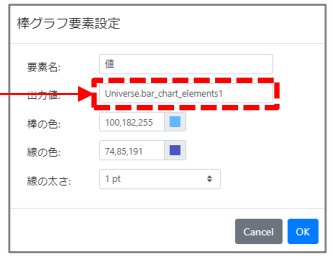
③ 出力パネル



④ 出力設定画面



⑤ 棒グラフ設定画面



⑥ 棒グラフ要素設定画面

Copyright© KOZOKEIKAKU ENGINEERING Inc. All Rights Reserved.

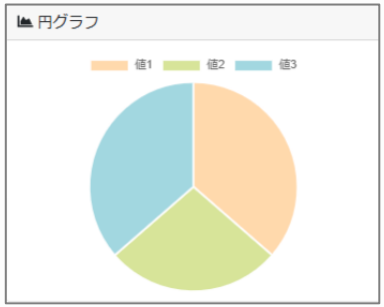
7. 円グラフ（1）

構造計画研究所
KOZO KEIKAKU ENGINEERING Inc.

円グラフを追加します

① モデルツリーの Universe 直下に変数を追加する
- 変数：pie_chart_elements

② Universe をクリックしてルールエディタを表示し、univ_step_end に下記の通りルールを追加する
- 0～5の整数値をランダムに3つ生成する



円グラフ



① モデルツリー

```
54  def univ_step_end(self):
55      Universe.ave_car_v = Universe.timechart_element_function0()
56      Universe.min_car_v = Universe.timechart_element_function1()
57
58      Universe.car_x = [c.x for c in Universe.jam_space.agents]
59      Universe.car_speed = [c.speed for c in Universe.jam_space.agents]
60
61      # レシピブック07.追加
62      Universe.bar_chart_elements1 = [random.randint(0, 5) for i in range(3)]
63      Universe.bar_chart_elements2 = [random.randint(0, 5) for i in range(3)]
64      Universe.pie_chart_elements = [random.randint(0, 5) for i in range(3)]
65      Universe.histogram_chart_elements = [random.randint(0, 100) for i in range(100)]
66
```

② ルールエディタ（Universe）

Copyright© KOZOKEIKAKU ENGINEERING Inc. All Rights Reserved.

7. 円グラフ（2）

③ 出力パネルの右上にマウスカーソルを移動し、設定アイコン をクリックする
④ 出力設定画面にて、円グラフ を選択してから 追加ボタン をクリックする
⑤ 円グラフ設定画面 が表示される。
⑥ 円グラフ要素リストの +アイコン をクリックして、円グラフ要素設定画面を表示し、以下の通り設定する

出力値：Universe.pie_chart_elements[0]

⑦ 同様に、円グラフ要素設定画面を表示し、以下の通り設定する

出力値：Universe.pie_chart_elements[1]
出力値：Universe.pie_chart_elements[2]

※ 出力値にはUniverse変数を指定でき、配列変数の場合は要素番号を指定することができる



③ 出力パネル



④ 出力設定画面



⑤ 円グラフ設定画面



⑥ 円グラフ要素設定画面

Copyright© KOZOKEIKAKU ENGINEERING Inc. All Rights Reserved.

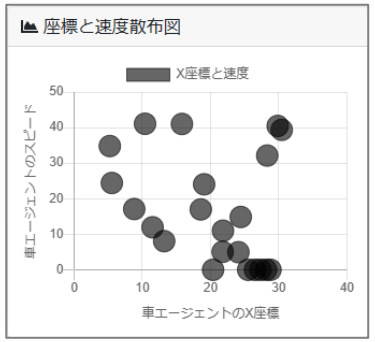
8. 散布図

散布図を追加します

① 出力パネルの右上にマウスカーソルを移動し、設定アイコン をクリックする
② 出力設定画面にて、散布図 を選択してから 追加ボタン をクリックする
③ 散布図設定画面にて、散布図要素リストの +アイコン をクリックして、散布図要素設定画面を表示し、以下の通り設定する

X座標：Universe.car_x
Y座標：Universe.car_speed

※ X座標、Y座標に同じ配列数の変数を指定することで、複数点をプロットできる



散布図



① 出力パネル



② 出力設定画面



③ 折れ線グラフ設定画面

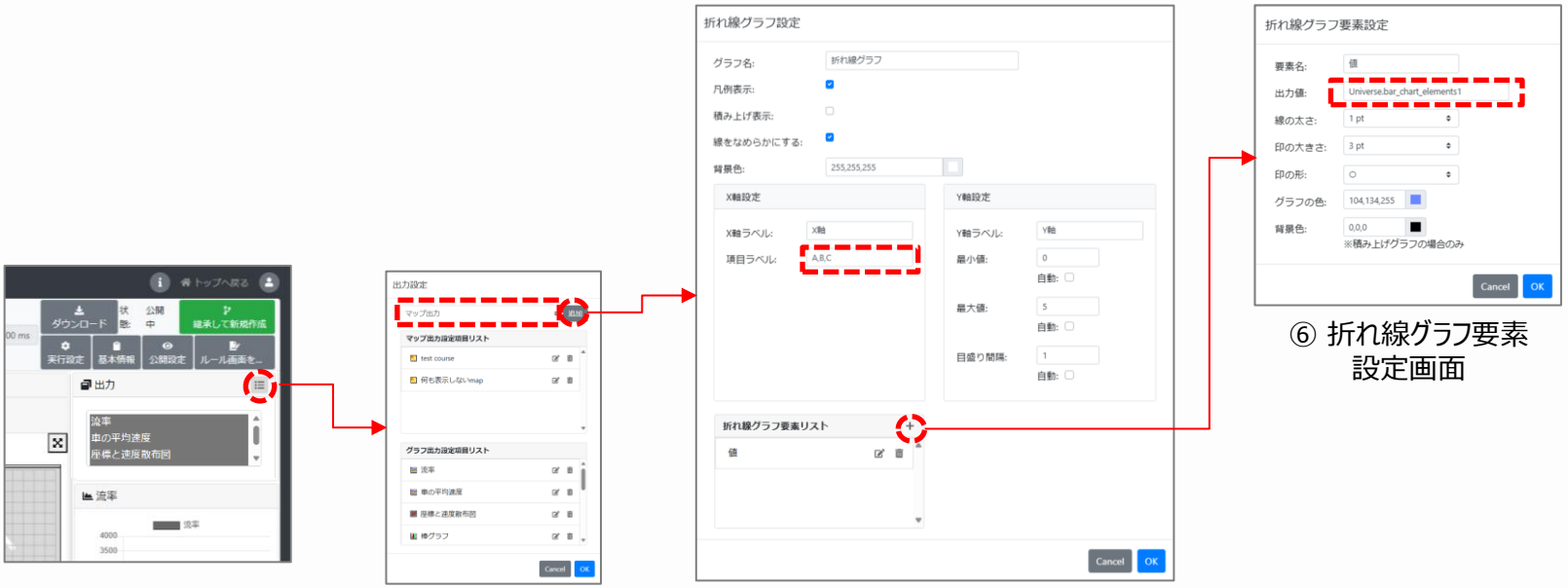


④ 散布図要素設定画面

Copyright© KOZOKEIKAKU ENGINEERING Inc. All Rights Reserved.

8. 散布図（2）

③ 出力パネルの右上にマウスカーソルを移動し、設定アイコン をクリックする
④ 出力設定画面にて、折れ線グラフ を選択してから 追加ボタン をクリックする
⑤ 折れ線グラフ設定画面にて、以下の通り設定する
　　X軸設定　項目ラベル：A,B,C　※ データ数だけカンマ区切りで項目ラベルを定義する
⑥ 折れ線グラフ要素リストの +アイコン をクリックして、折れ線グラフ要素設定画面を表示し、以下の通り設定する
　　出力値：Universe.bar_chart_elements1



③ 出力パネル　④ 出力設定画面　⑤ 折れ線グラフ設定画面　⑥ 折れ線グラフ要素設定画面

Copyright© KOZOKEIKAKU ENGINEERING Inc. All Rights Reserved.

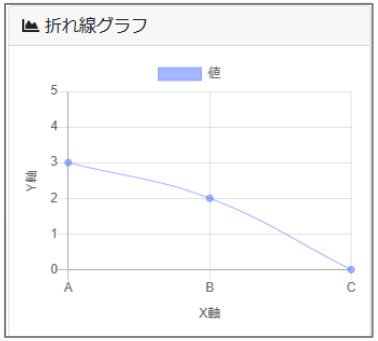
9. 折れ線グラフ（1）

構造計画研究所
KOZO KEIKAKU ENGINEERING Inc.

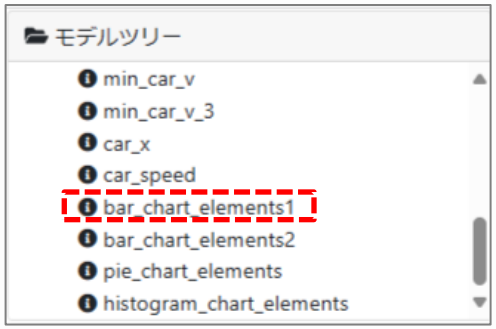
折れ線グラフを追加します

① モデルツリーの Universe 直下に変数を追加する
- 変数：bar_chart_elements1

② Universe をクリックしてルールエディタを表示し、univ_step_end に下記の通りルールを追加する
- 0～5の整数値をランダムに3つ生成する



折れ線グラフ



① モデルツリー

```
def univ_step_end(self):
    Universe.ave_car_v = Universe.timechart_element_function0()
    Universe.min_car_v = Universe.timechart_element_function1()

    Universe.car_x = [c.x for c in Universe.jam_space.agents]
    Universe.car_speed = [c.speed for c in Universe.jam_space.agents]

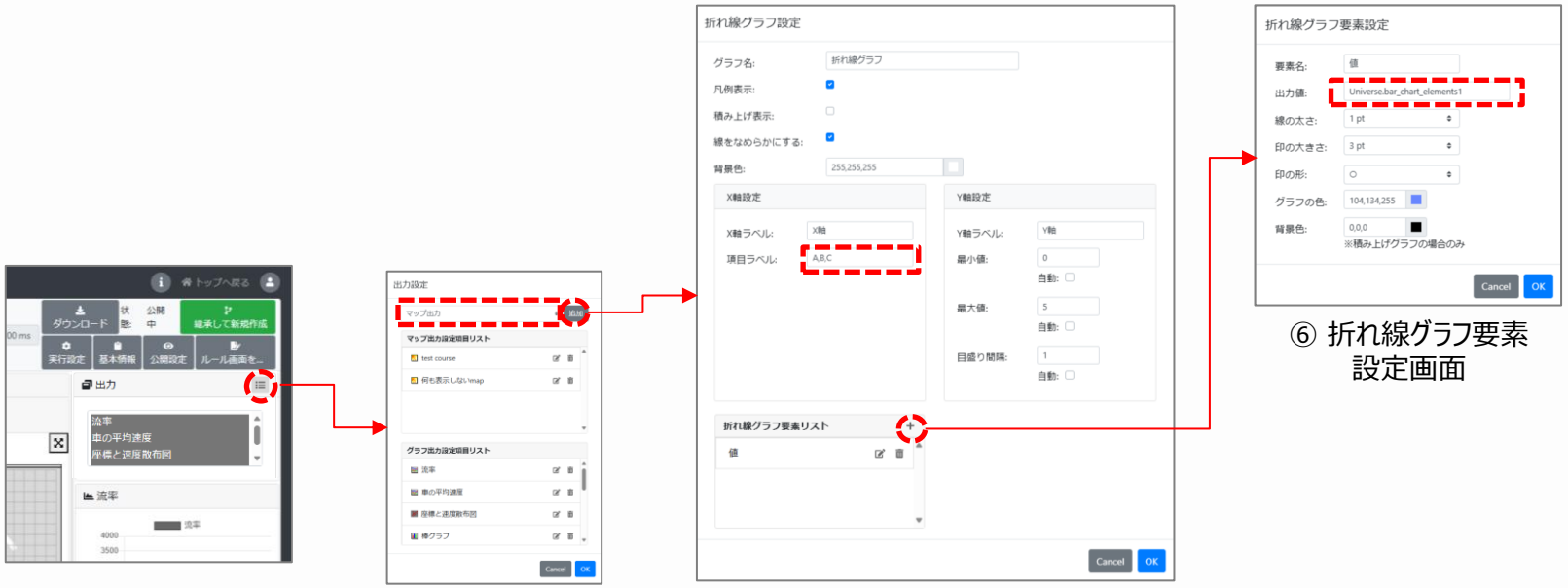
    # レシピブック07.追加
    Universe.bar_chart_elements1 = [random.randint(0, 5) for i in range(3)]
    Universe.bar_chart_elements2 = [random.randint(0, 5) for i in range(3)]
    Universe.pie_chart_elements = [random.randint(0, 5) for i in range(3)]
    Universe.histogram_chart_elements = [random.randint(0, 100) for i in range(100)]
```

② ルールエディタ（Universe）

Copyright© KOZOKEIKAKU ENGINEERING Inc. All Rights Reserved.

9. 折れ線グラフ（2）

③ 出力パネルの右上にマウスカーソルを移動し、設定アイコン をクリックする
④ 出力設定画面にて、折れ線グラフ を選択してから 追加ボタン をクリックする
⑤ 折れ線グラフ設定画面を以下の通り設定する
　　X軸設定　項目ラベル：A,B,C　※ データ数だけカンマ区切りで項目ラベルを定義する
⑥ 折れ線グラフ要素リストの +アイコン をクリックして、折れ線グラフ要素設定画面を表示し、以下の通り設定する
　　出力値：Universe.bar_chart_elements1



③ 出力パネル　④ 出力設定画面　⑤ 折れ線グラフ設定画面　⑥ 折れ線グラフ要素設定画面

Copyright© KOZOKEIKAKU ENGINEERING Inc. All Rights Reserved.

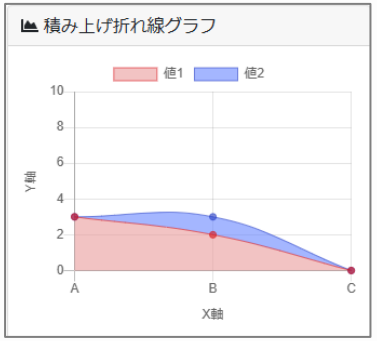
10. 積み上げ折れ線グラフ（1）

構造計画研究所
KOZO KEIKAKU ENGINEERING Inc.

積み上げ折れ線グラフを追加します

① モデルツリーの Universe 直下に変数を追加する
- 変数：bar_chart_elements1
- 変数：bar_chart_elements2

② Universe をクリックしてルールエディタを表示し、univ_step_end に下記の通りルールを追加する
- 0～5の整数値をランダムに3つ生成する



積み上げ折れ線グラフ



① モデルツリー

```
54  def univ_step_end(self):
55      Universe.ave_car_v = Universe.timechart_element_function0()
56      Universe.min_car_v = Universe.timechart_element_function1()
57
58      Universe.car_x = [c.x for c in Universe.jam_space.agents]
59      Universe.car_speed = [c.speed for c in Universe.jam_space.agents]
60
61      # レシピブック07.追加
62      Universe.bar_chart_elements1 = [random.randint(0, 5) for i in range(3)]
63      Universe.bar_chart_elements2 = [random.randint(0, 5) for i in range(3)]
64      Universe.pie_chart_elements = [random.randint(0, 5) for i in range(3)]
65      Universe.histogram_chart_elements = [random.randint(0, 100) for i in range(100)]
66
```

② ルールエディタ（Universe）

Copyright© KOZOKEIKAKU ENGINEERING Inc. All Rights Reserved.

10. 積み上げ折れ線グラフ（2）

構造計画研究所
KOZO KEIKAKU ENGINEERING Inc.

③ 出力パネルの右上にマウスカーソルを移動し、設定アイコン をクリックする
④ 出力設定画面にて、折れ線グラフ を選択してから 追加ボタン をクリックする
⑤ 折れ線グラフ設定画面を以下の通り設定する
　　積み上げ表示：チェックON
　　X軸設定　項目ラベル：A,B,C　※ データ数だけカンマ区切りで項目ラベルを定義する
⑥ 折れ線グラフ要素リストの +アイコン をクリックして、折れ線グラフ要素設定画面を表示し、以下の通り設定する
　　出力値：Universe.bar_chart_elements1
⑦ 同様に、折れ線グラフ要素設定画面を表示し、以下の通り設定する
　　出力値：Universe.bar_chart_elements2



③ 出力パネル



④ 出力設定画面



⑤ 折れ線グラフ設定画面



⑥ 折れ線グラフ要素
設定画面

Copyright© KOZOKEIKAKU ENGINEERING Inc. All Rights Reserved.

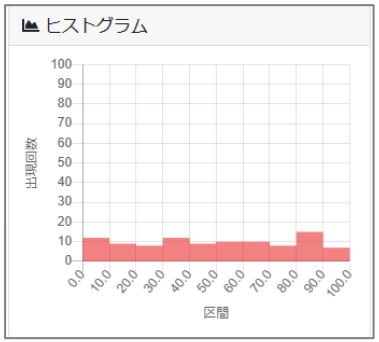
11. ヒストグラム（1）

構造計画研究所
KOZO KEIKAKU ENGINEERING Inc.

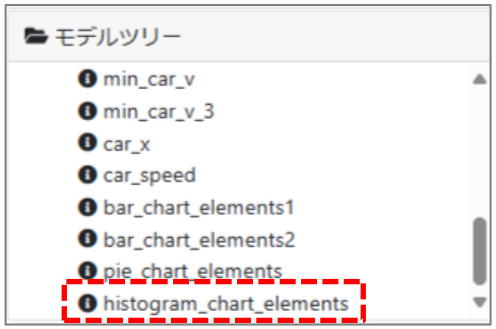
ヒストグラムを追加します

① モデルツリーの Universe 直下に変数を追加す
- 変数：histogram_chart_elements

② Universe をクリックしてルールエディタを表示し、univ_step_end に下記の通りルールを追加する
- 0～100の整数値をランダムに100個生成する



ヒストグラム



① モデルツリー

```
54 def univ_step_end(self):
55     Universe.ave_car_v = Universe.timechart_element_function0()
56     Universe.min_car_v = Universe.timechart_element_function1()
57
58     Universe.car_x = [c.x for c in Universe.jam_space.agents]
59     Universe.car_speed = [c.speed for c in Universe.jam_space.agents]
60
61     # レシピブック07.追加
62     Universe.bar_chart_elements1 = [random.randint(0, 5) for i in range(3)]
63     Universe.bar_chart_elements2 = [random.randint(0, 5) for i in range(3)]
64     Universe.pie_chart_elements = [random.randint(0, 5) for i in range(3)]
65     Universe.histogram_chart_elements = [random.randint(0, 100) for i in range(100)]
66
```

② ルールエディタ（Universe）

Copyright© KOZOKEIKAKU ENGINEERING Inc. All Rights Reserved.

11. ヒストグラム（2）

③ 出力パネルの右上にマウスカーソルを移動し、設定アイコン をクリックする
④ 出力設定画面にて、 ヒストグラム を選択してから 追加ボタン をクリックする
⑤ ヒストグラム設定画面を以下の通り設定する

- X軸設定
  区間数：10
  区間の指定：0～100
- 出力値：Universe.histogram_chart_elements



③ 出力パネル　④ 出力設定画面　⑤ ヒストグラム設定画面

Copyright© KOZOKEIKAKU ENGINEERING Inc. All Rights Reserved.

12. 値出力

値出力を追加します

① 出力パネルの右上にマウスカーソルを移動し、設定アイコン をクリックする
② 出力設定画面の 出力形式選択リスト から 値出力 を選択してから追加ボタン をクリックする
③ 値出力設定画面の値出力要素リストの ＋アイコン をクリックする
④ 値要素設定画面を以下の通り設定する
　　出力値：Universe.flow_value



値出力



① 出力パネル　② 出力設定画面　③ 値出力設定画面　④ 値出力要素設定画面

Copyright© KOZOKEIKAKU ENGINEERING Inc. All Rights Reserved.