

# ものづくり創成実習Ⅱ 3班発表

---

# 実習の内容

---

(1) GAによる最適速度データの獲得

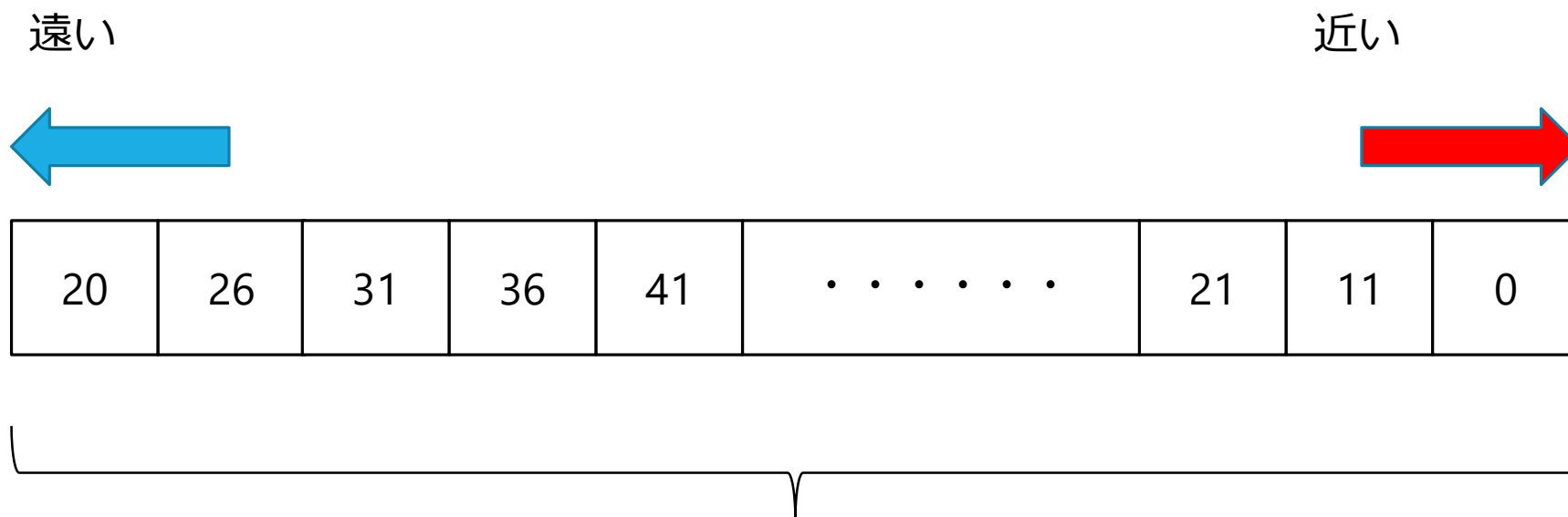
(2) LEGOロボットの知的制御

## (1) GAによる最適速度データの獲得

---

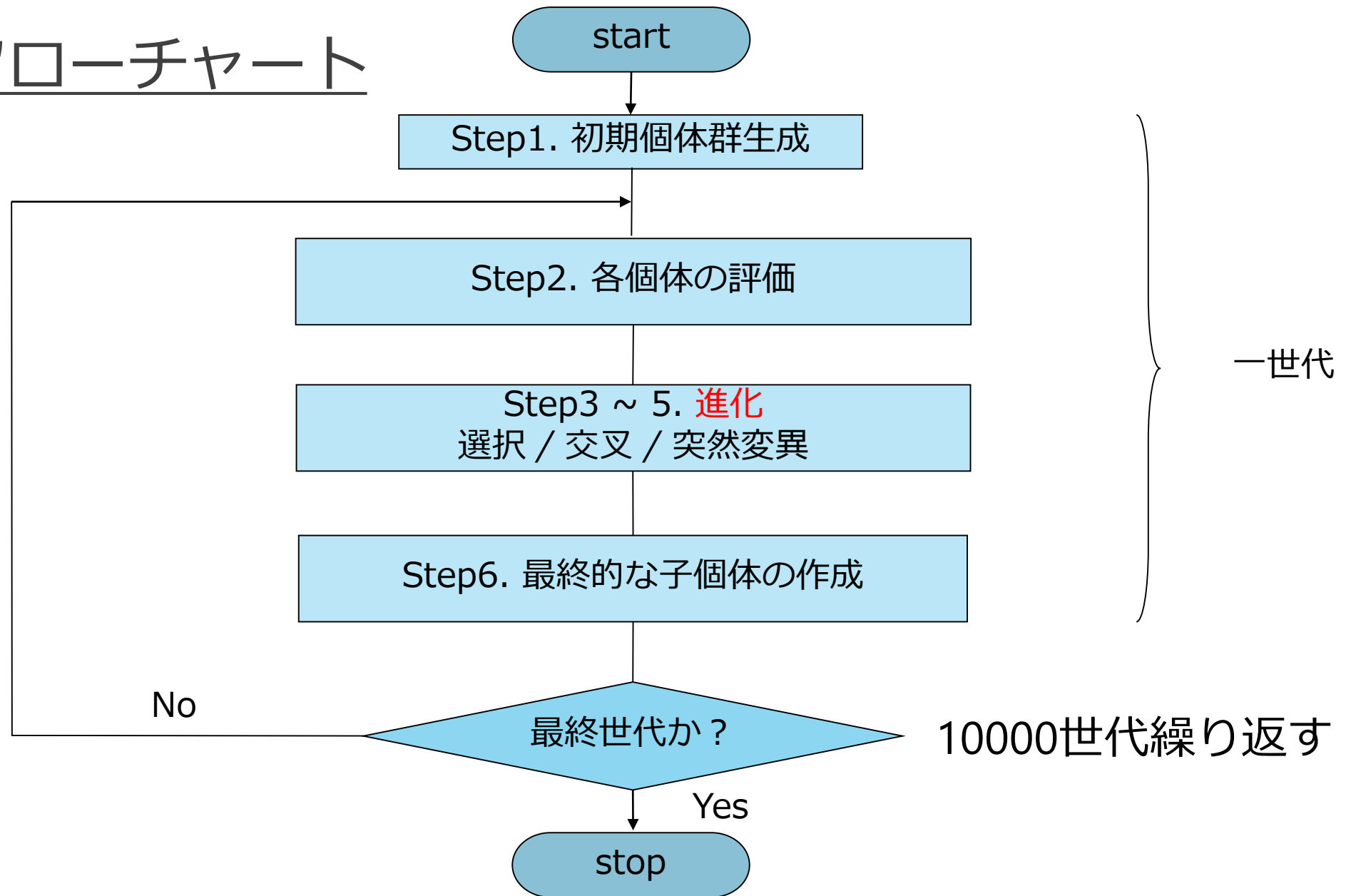
# (1) GAにおける個体表現

距離に対するスピードをGAで獲得する際の遺伝子型



**遺伝子長 : 30**

# (1) GAのフローチャート



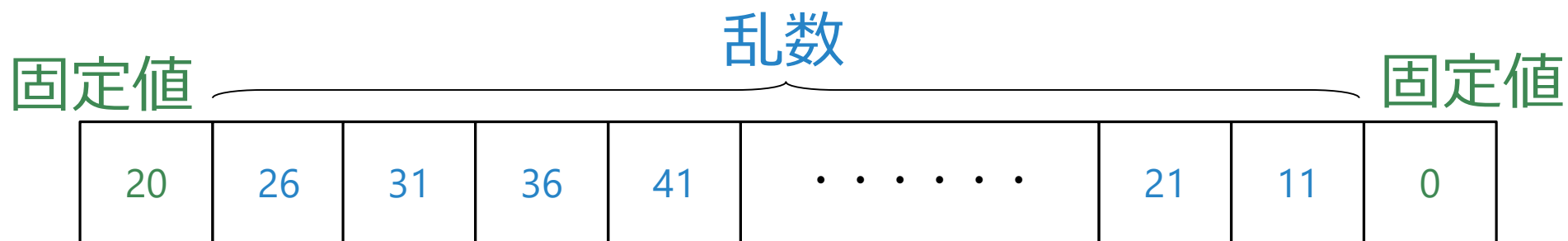
# (1) Step1.初期個体群の生成

- ・ 遺伝子長**30**の個体を51個生成する。

- ・ 遺伝子座

1～28番目 : 1 ~ 100の乱数

0・29番目 : 20 と 0 (固定値)



遺伝子長 : 30

# (1) Step2. 各個体の評価

---

求める最適速度データ：

「できるだけ速く」

「滑らかな速度変化」

という二つの目的を同時に満たすような速度データ

⇒ 「多目的最適化問題」

# (1) Step2. 各個体の評価

---

評価方法:

以下の適合度関数を設計し、適合度を評価

$$\text{適合度} = \text{速度合計値} - (\text{速度変化})^2$$

目的関数：できるだけ速く

目的関数：滑らかな速度変化



# (1) Step3. 選択

---

- **エリート選択**  
： その世代において**最も適合度の高い個体**を1つ保存
- **トーナメント選択**  
： 計51個の個体から、計50回、**トーナメントサイズ2**の  
トーナメント方式で選択  
(比較する2つの個体の選び方はランダム)  
  
⇒ 適合度の高い方の遺伝子情報を選択

# (1) Step4. 交叉

---

- ・ 一様交叉

親 : Step3で選択した個体の隣り合う二組

**交叉率 : 0.5**

各親個体の遺伝子毎にランダムに交叉するかしないかを決定。

⇒計25回 (=50個分の子個体分) 実行

# (1) Step5. 突然変異

---

- ・ **突然変異確率 : 0.01**

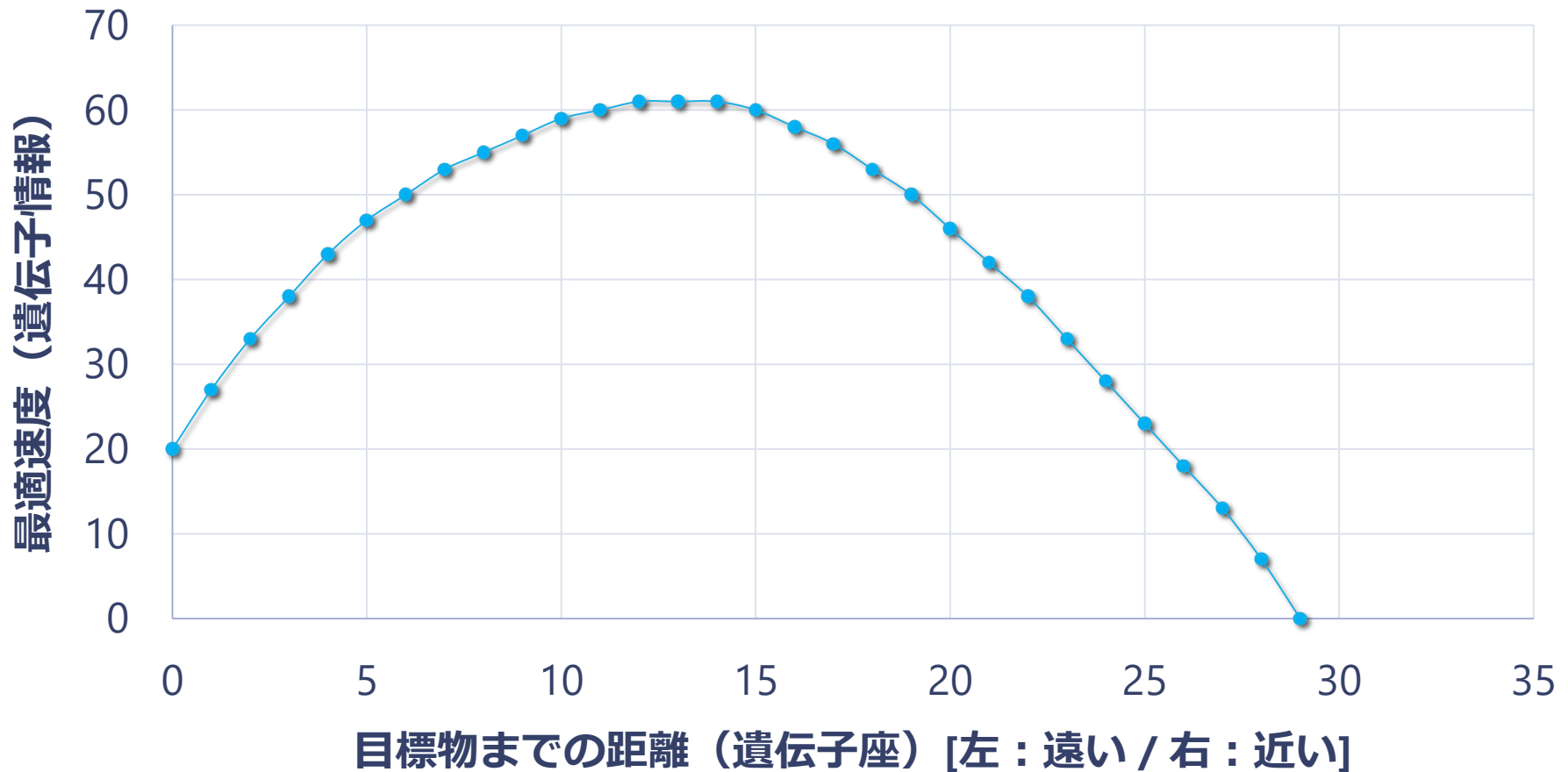
交叉済みの50個の子個体に対して突然変異を行う

1 ~ 28番目の遺伝子座に対して、

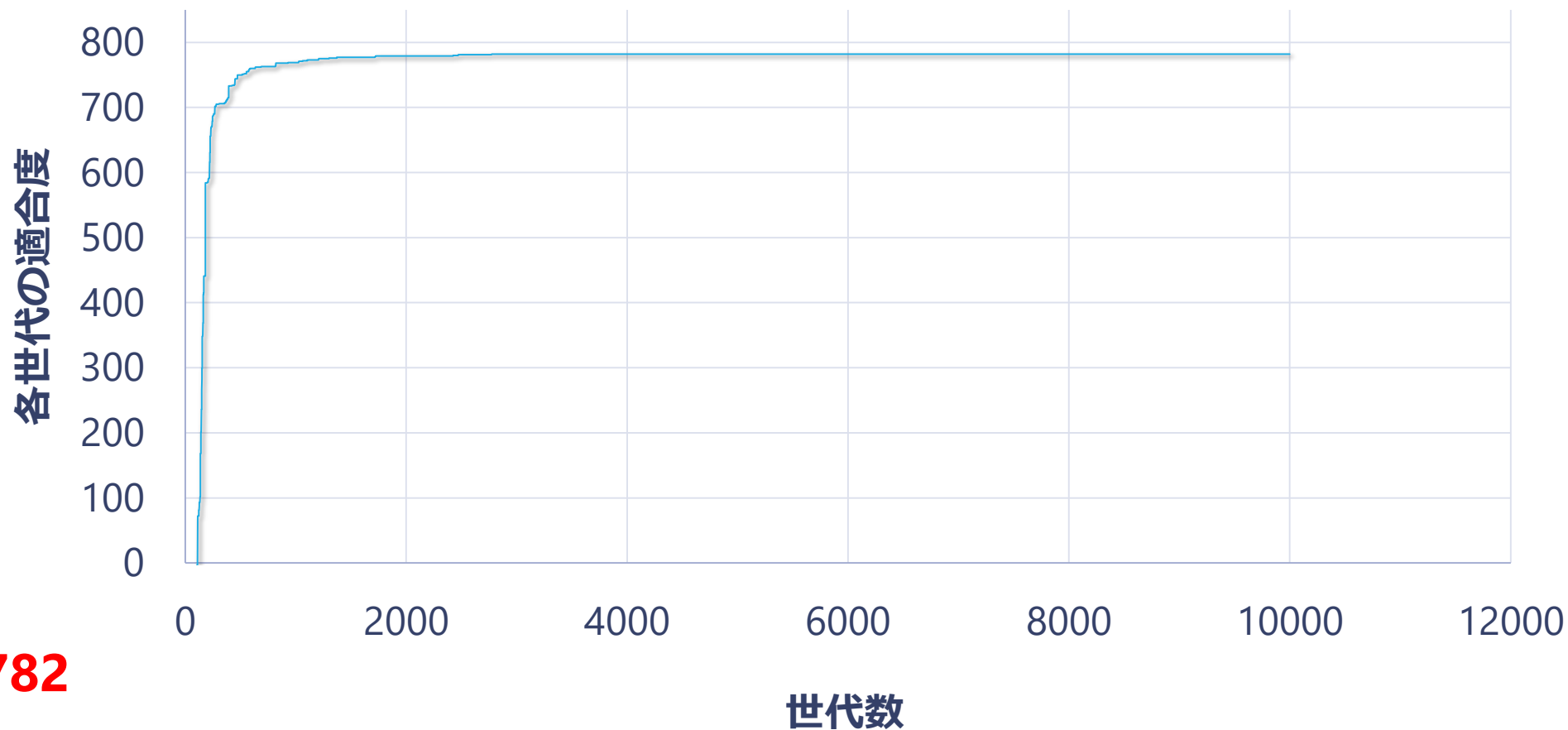
「1 ~ 100」の乱数に変異

(平均的には**1個**の遺伝子座に対して突然変異が適用)

# (1) 実行結果\_\_最適速度データ



# (1) 実行結果\_\_G Aの進化曲線



適合度 : 782

# (1) より速い速度データを得るには？

---

考えられる手段として、

- ・ 適応度関数に制約条件を設ける  
(制約違反の場合、ペナルティを与える)
- ・ 適応度関数の設計を工夫する  
(目的関数に係数を付ける)

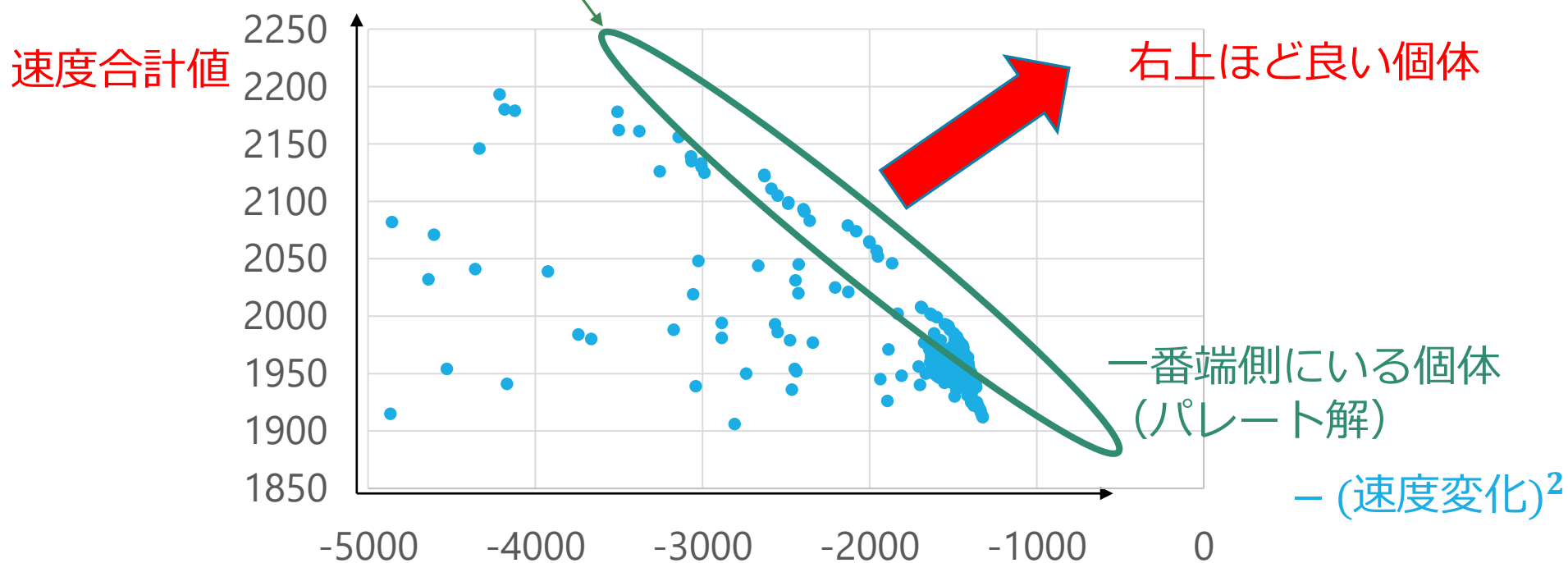
# (1) より速い速度データを得るには？

- 適合度関数：適合度 = 速度合計値 - (速度変化)<sup>2</sup>
- 目的関数「速度合計値」「-(速度変化)<sup>2</sup>」を同時に最適化することは困難  
(片方が良くなるともう片方が悪くなる)
- 目的関数である「速度合計値」「-(速度変化)<sup>2</sup>」をそれぞれ分けて評価することで、関係性（トレードオフ関係）を見つける

# (1) より速い速度データを得るには？

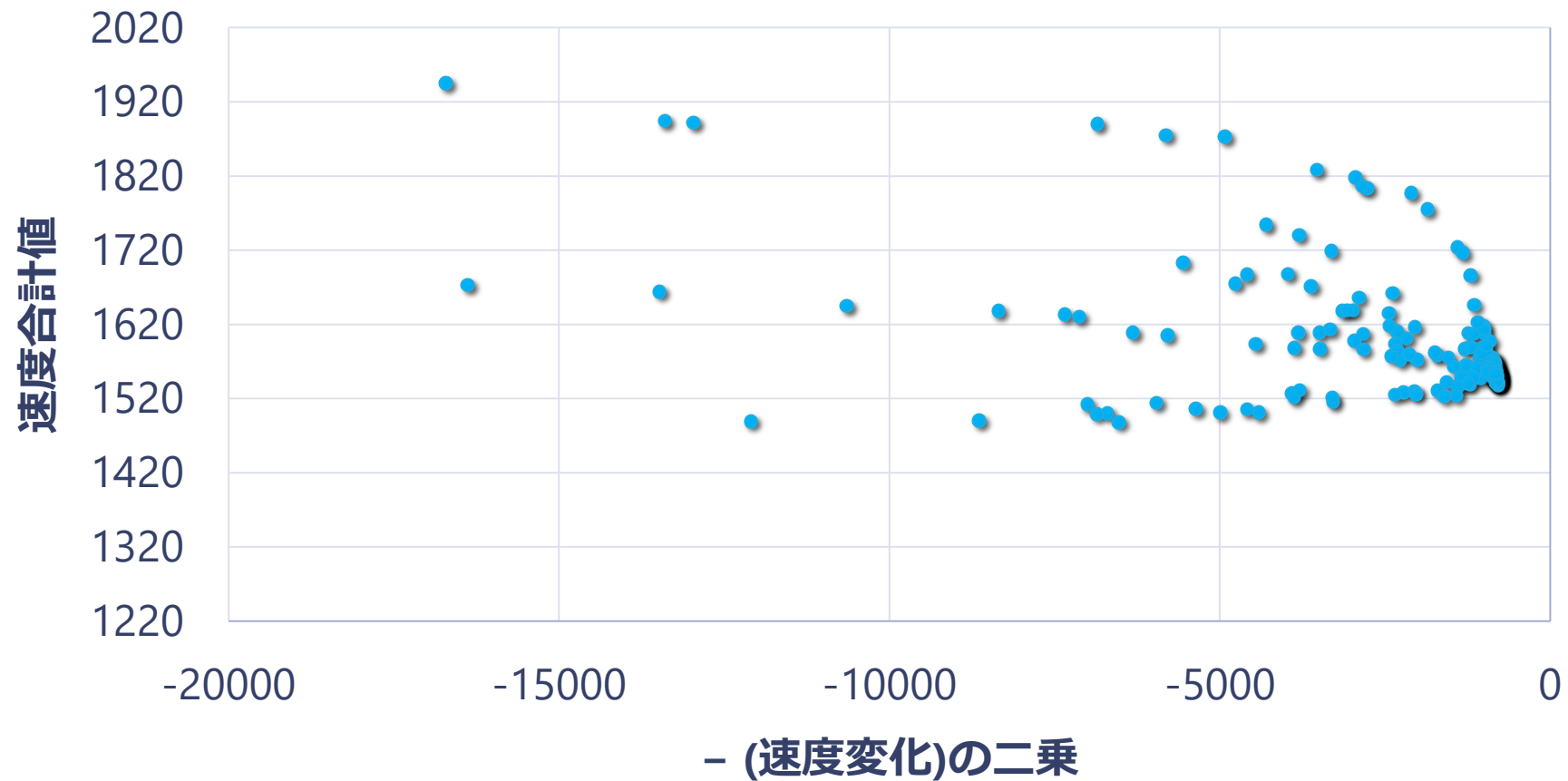
- 全個体中で、「**速度合計値**」、「 **$-(\text{速度変化})^2$** 」のどちらともが**他よりも低くない個体**を次世代に残す

例)

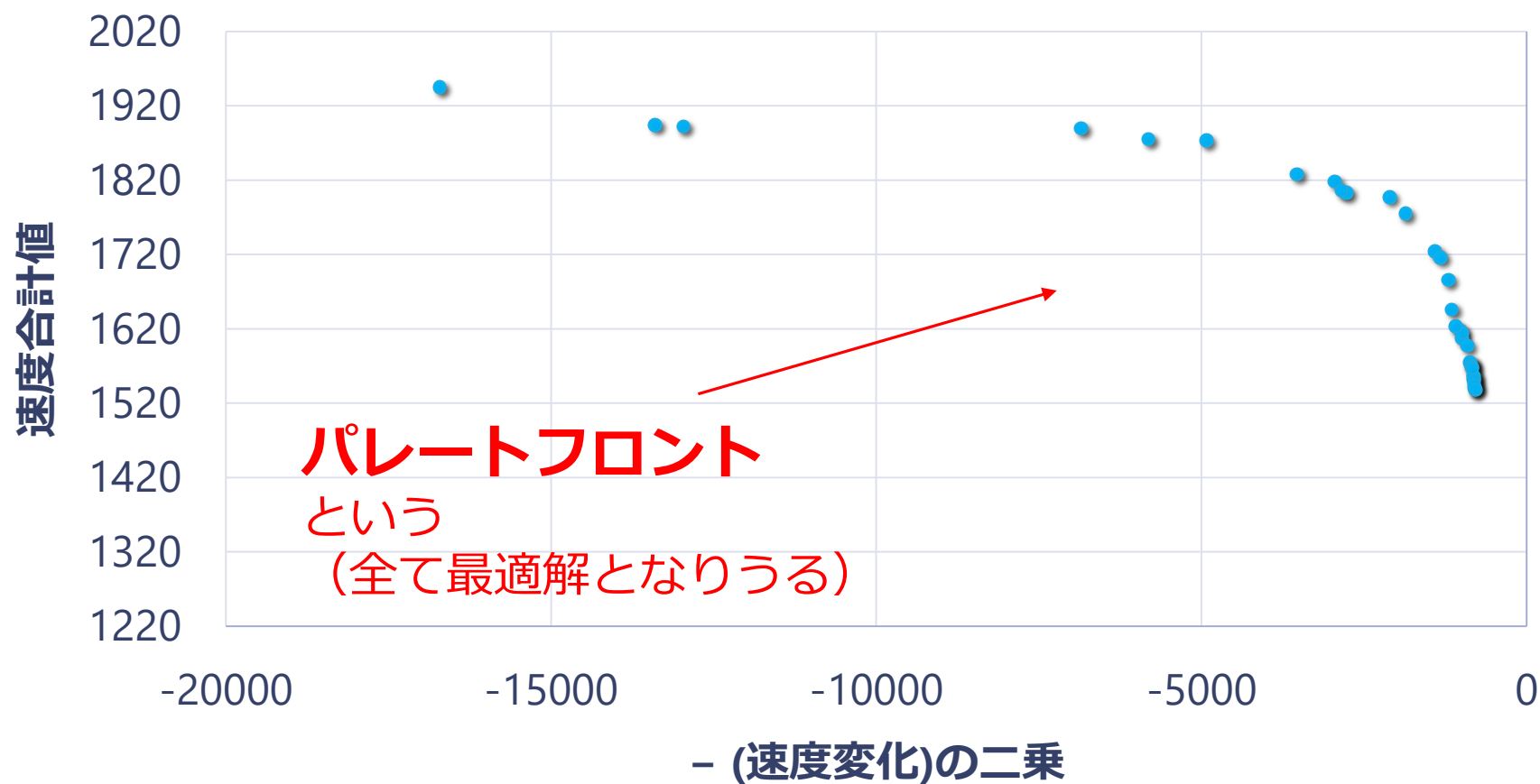




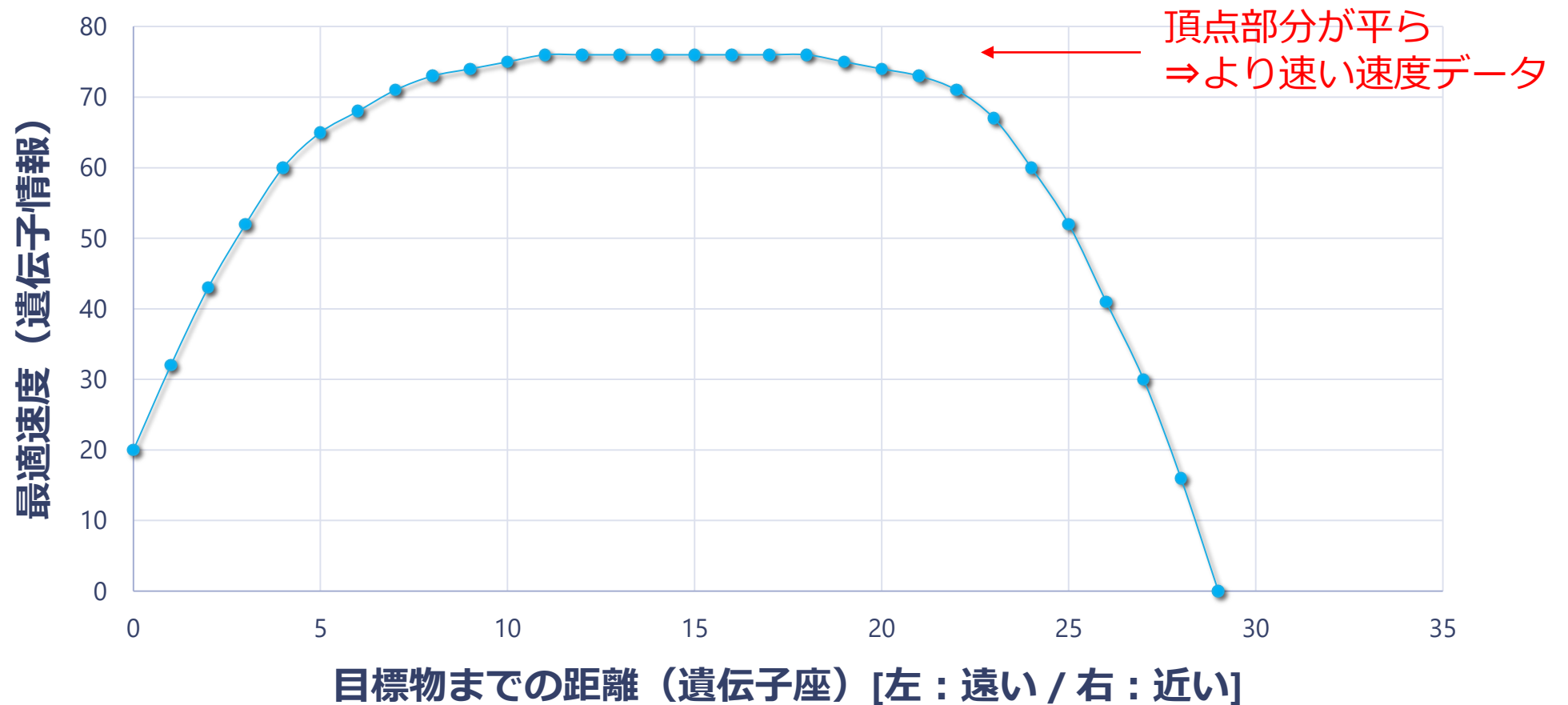
# (1) 得られた個体の分布



# (1) 得られた端側にいる個体の分布



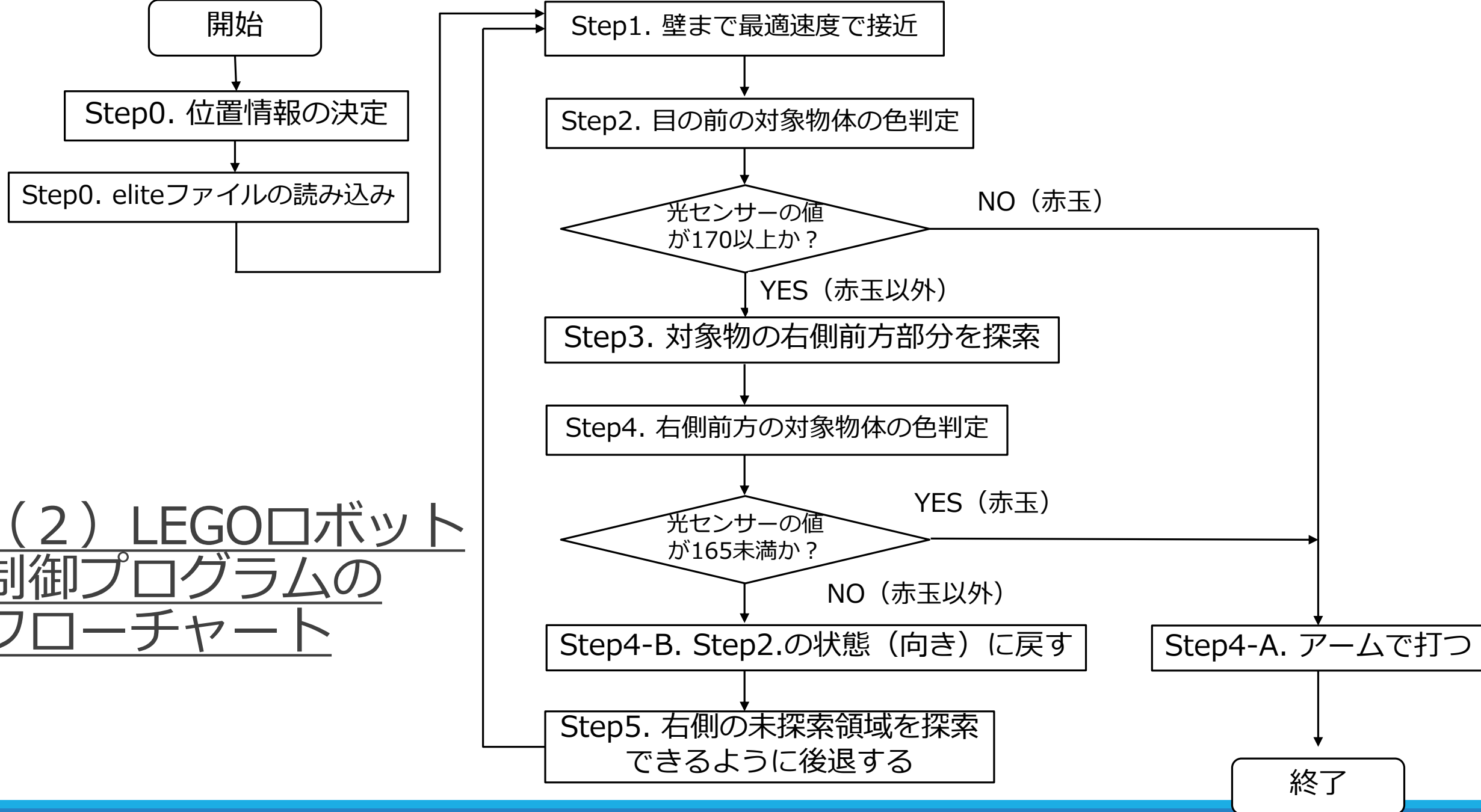
# (1) 速度合計値1800以下でペナルティを与えた場合の最適速度データ



## (2) LEGOロボットの知的制御

---

## (2) LEGOロボット 制御プログラムの フローチャート



## ( 2 ) Step 0 . 位置情報の決定

---

- ・ 超音波センサーの取り得る距離の数値を**30**等分  
（ 1 区間 : 8 ）
- ・ 30分割した値（位置情報）を境に、  
超音波センサーの値が、その**位置情報の値以上**であれば、対応する最適速度データで走行

## (2) Step 0 . 位置情報の決定

超音波センサーの値  
(位置情報) :

240	232	224	216	208	.....	24	16	8
-----	-----	-----	-----	-----	-------	----	----	---



最適速度データ

20	26	31	36	41	.....	21	11	0
----	----	----	----	----	-------	----	----	---

超音波センサーの値が  
各位置情報の値以上なら、  
対応する最適速度で走行する

遠い

30分割

近い



## ( 2 ) Step 0 . 位置情報の決定

---

- ・ 壁間際での超音波センサーの値 : 「24」

⇒超音波センサーの値が「25」の時、  
最適速度データの最終速度の一つ前の速度で走行し、  
**超音波センサーの値が「24」となった瞬間、  
最終速度 (= 0) となればよい**

⇒**位置情報の末端の一つ前**の値が「25」となるように、  
30等分した位置情報に「9」を足せばよい。



240を30分割した位置情報における壁間際の値が、  
壁間際の超音波センサーの値（= **24**）以下の時、  
最終速度となるように、各位置情報に「**9**」を足している。

超音波センサーの値  
（位置情報）：

255	247	239	231	223	.....	33	25	17
-----	-----	-----	-----	-----	-------	----	----	----

最適速度データ

↓	↓	↓	↓	↓		↓	↓	↓
20	26	31	36	41	.....	21	11	0

超音波センサーの値が  
各位置情報の値以上なら、  
対応する最適速度で走行する

遠い

30分割

近い



## (2) Step 1 ~ 5.

- ・ 赤玉を探索するまで、フローチャートに従って探索し続ける。

- ・ Step1. 壁間際まで最適速度で移動

- ・ Step2. 色判定

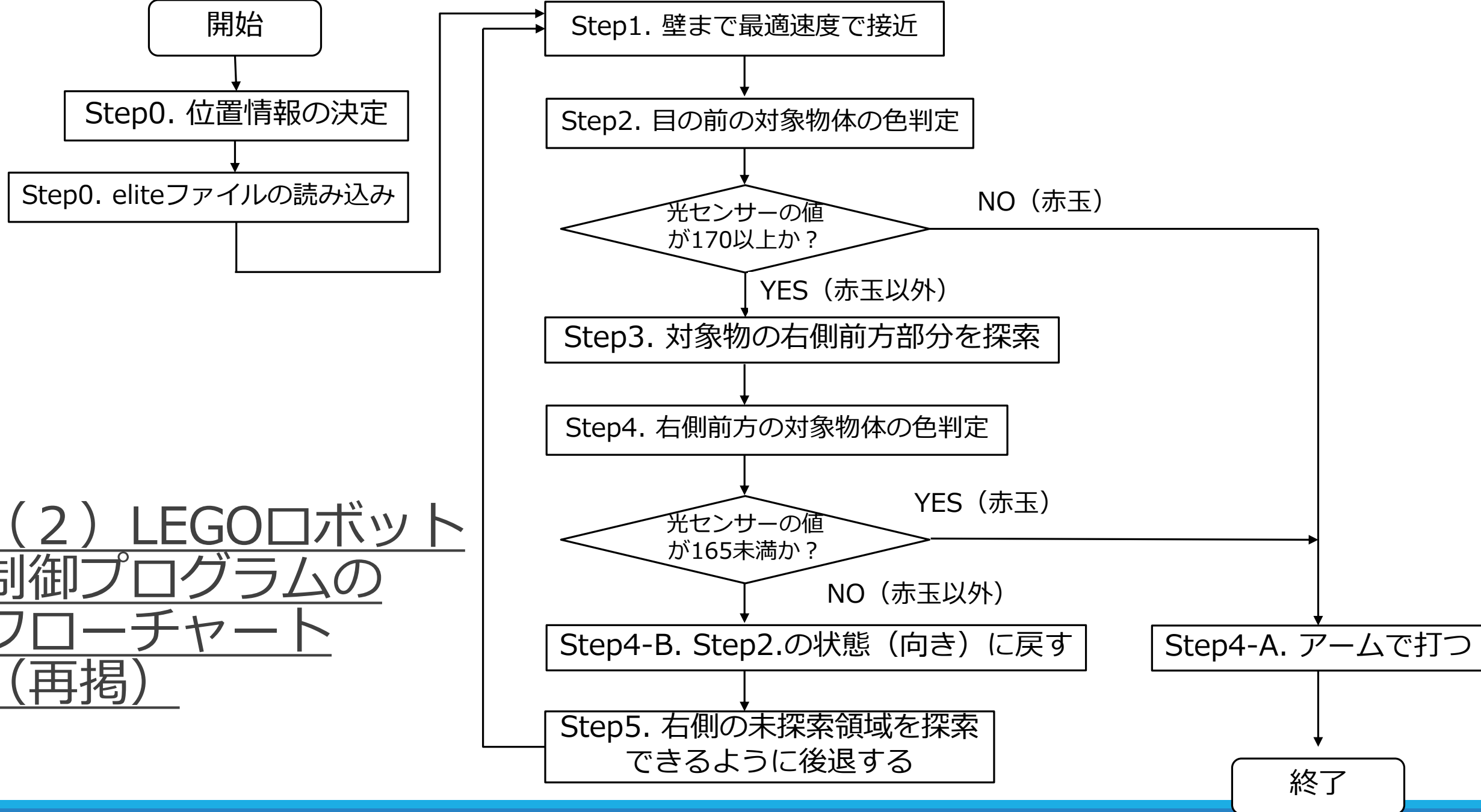
- ・ ⇒赤玉以外⇒右側領域も探索⇒再度色判定

⇒赤玉

⇒赤玉以外⇒探索続行

⇒赤玉⇒アームで打つ<sub>(終)</sub>

アーム出力値 : 40



## (2) LEGOロボット 制御プログラムの フローチャート (再掲)

# • LEGOロボットの知的制御の工夫

---

- 一度の探索において、一度の色判定だけで次の探索ステップに移らず、右側領域部分も加えて探索する点

⇒赤玉の探索見逃しを防ぎ、赤玉探索効率を向上

- 位置情報を用い、30分割することで、室内の明るさが変わることによる値のずれも微調整することができる点。

# 成功動画

---



# まとめ

---

- 「GA」を用い、「最適速度データ」を求めることができた。
- 「MOGA」を作成することで、パレート曲線の抽出にも成功した。
- LEGOロボットの効率的な知的制御を実現できた。