

遺伝的アルゴリズムに基づく 鉄道ダイヤの生成自動化

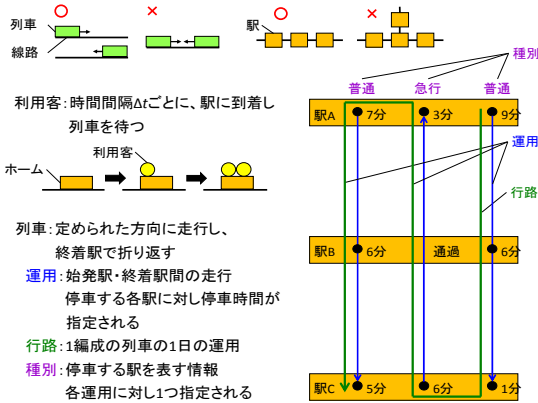
背景

- 鉄道ダイヤの乱れは多くの人に影響を及ぼすが、ダイヤの乱れを完全になくすことは難しい
- 運休・順序変更等の運行計画の変更は、人の手で行われるため多大な時間を要する
- 遺伝的アルゴリズムを用いて鉄道ダイヤの自動生成を試みる

鉄道ダイヤ最適化問題

想定する鉄道交通システムのモデル

線路: 複々線、分岐は考えない



不満足

全利用客の待ち時間に関する不満を数値化した値
不満足 $d = A$
 A : 全利用客の列車を待つ時間の総和

最適化目標
不満足の最小化

鉄道ダイヤ生成手法

遺伝的アルゴリズムを適用

解の表現

種別: 停車・通過を表す情報を駅の数だけ並べる
行路: 運用の数だけ並べる
種別の番号を1つ
停車時間の情報を駅の数だけ
営業時間内で走行できない分は無視



鉄道シミュレータでの評価

- 時間間隔 Δt ごとの状態を求める
各列車の位置
各駅のホームで列車を待つ利用客の人数
 - 仕様
列車は一定の速さで走行する
端の駅で折り返す
列車を待つ人は、列車が来たら全員乗車する
- 各利用客の列車の待ち時間から不満度を求める

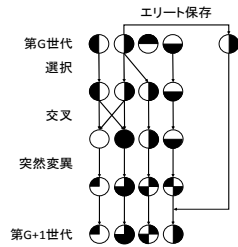
遺伝的操作

- 選択: 適応度比例戦略 (ルーレット選択)
適応度関数 $F = \frac{|d - d_{max}|}{d_{max}}$
 d : 各個体の不満足度
 d_{max} : その世代に含まれる不満足度の最大値
- 交叉: 一様交叉
- 突然変異: 解の構成要素を突然変異率の割合で変化させる

全体の流れ

Simple GA

- 解を選択・淘汰する
 - 解を2つで1組とし、交叉率の割合で交叉を行う
 - 全ての個体に突然変異を行う
- エリート保存戦略
対象とする世代の最良解と、次世代の最悪解を入れ替える



鉄道ダイヤ最適化問題への適用

鉄道ダイヤの自動生成実験

列車
行路数(列車数): 12
運用数: 40
種別数: 6+各駅停車
営業時間: 19時間
利用客: 1,216,380人
世代交代モデル
Simple GA
Elite GA (Simple GA
+エリート保存戦略)

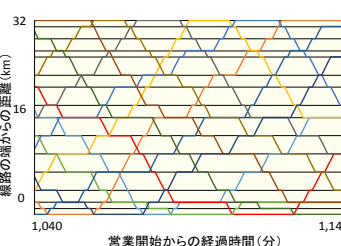
駅の番号	駅の位置 (km)	利用客の増加割合(人/分)		初期列車数	
		上り	下り	上り	下り
1	0	-	470	-	3
2	1	3	3	0	0
3	2	32	32	0	0
4	4	8	9	0	0
5	7	16	17	0	0
6	10	26	26	0	0
7	13	26	27	0	0
8	16	48	48	3	3
9	18	15	14	0	0
10	21	7	7	0	0
11	23	17	17	0	0
12	26	10	9	0	0
13	27	18	18	0	0
14	29	11	11	0	0
15	31	6	5	0	0
16	32	111	-	3	-

パラメータ

- 遺伝的アルゴリズム
個体数: 110
交叉率: 0.7, 突然変異率: 5.0×10^{-4}
- 鉄道シミュレータ
時間間隔 Δt : 1分
列車の走行速度: 1km/分

Elite GAにより得たダイヤ

- 不満足度が最適化前の約36%になった
- 列車がまばらに走行している
- 駅における列車の停車時間が長い



停車時間の短縮

不満度の再定義

停車時間を短くするため、停車時間の不満を考慮

$$d = A + w \times B$$

A : 全利用客の列車を待つ時間の総和

B : 列車の各駅に対する停車時間 - 最小停車時間

w : 停車時間に関する不満の重み

$w = 4,800$ の場合のダイヤ

- 各駅における停車時間が1~2分になった
- 停車時間が1運用あたり約40分短縮された
- 短い時間で利用客を先の駅に運んだ

