講義「人工知能」
第10回 進化型計算
意思決定システムの進化的設計

北海道大学 大学院情報科学研究院
情報理工学部門 複合情報工学分野 調和系工学研究室
准教授 山下倫央
http://harmo-lab.jp
tomohisa@ist.hokudai.ac.jp
2024年5月9日(木)

harmo-lab.jp
調和系工学研究室

ダーウィンの進化論

- 地球上に生活している150万種以上の様々な格好をした膨大な生物種がいるが，これらの起源はいったい何なのか？
- さらに生物たちはどうしてこんなに不思議な格好をしているのだろうか？

チャールズ・ダーウィン
(1809～1882)

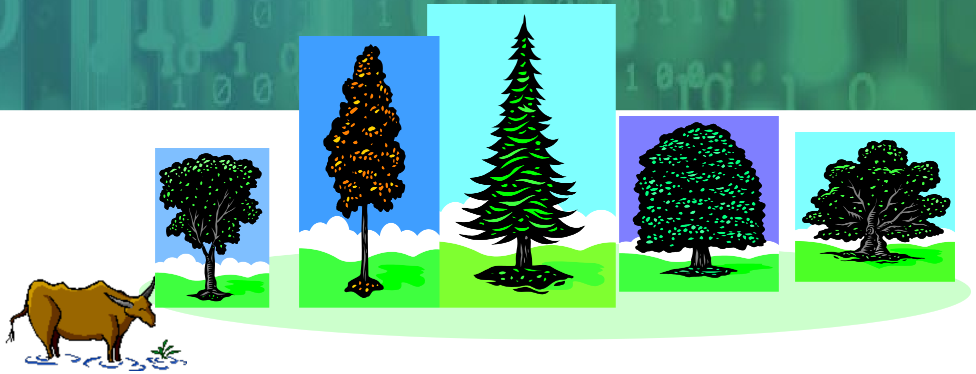
イギリスの博物学者．エジンバラ大学医学部，ケンブリッジ大学神学部に学ぶ．1859年，自然淘汰による進化論「種の起源」を発表した．

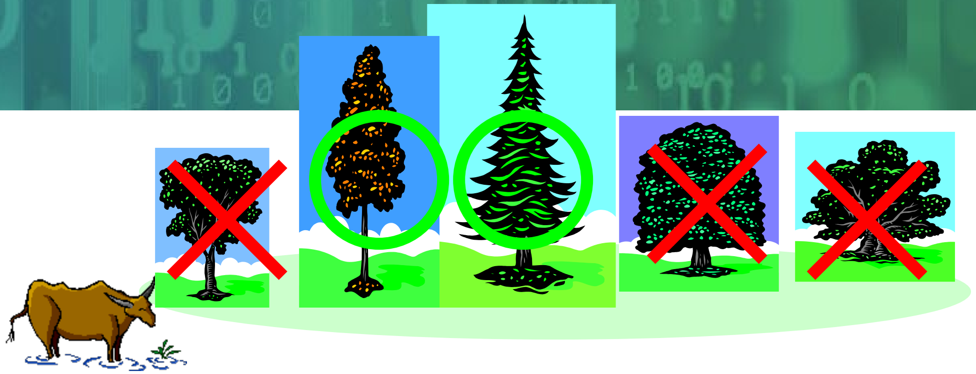
ガラパゴス諸島を含む観察に基づく結論

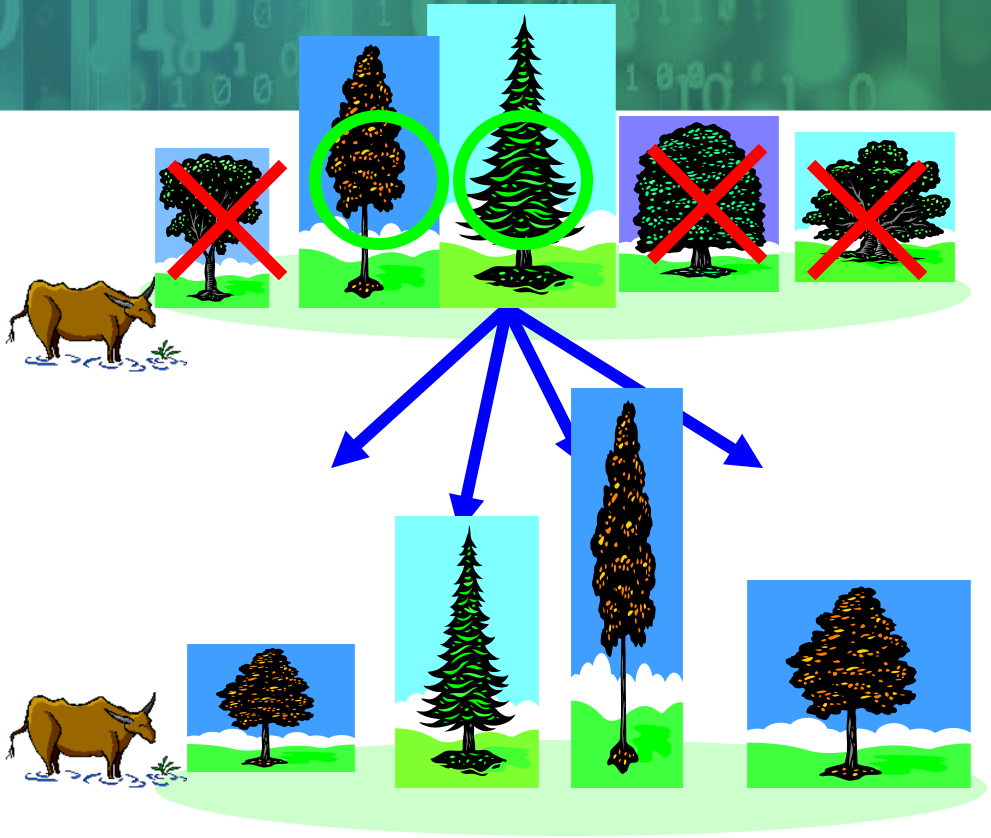
進化論

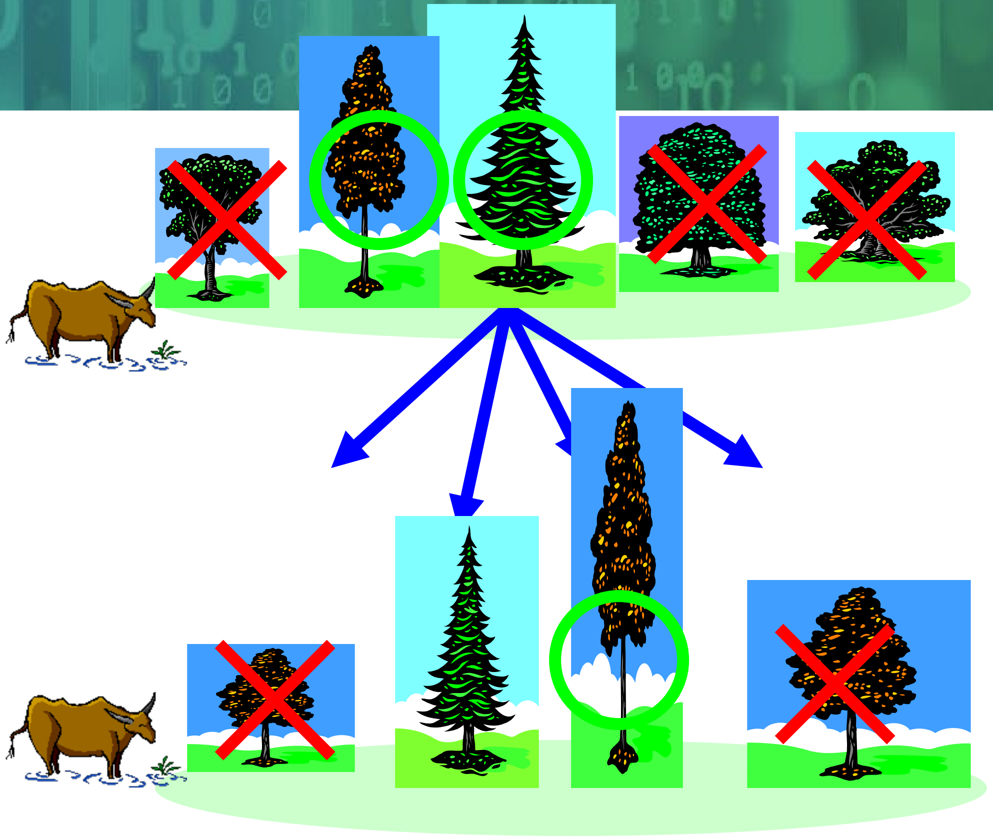
ダーウィンの進化の概念

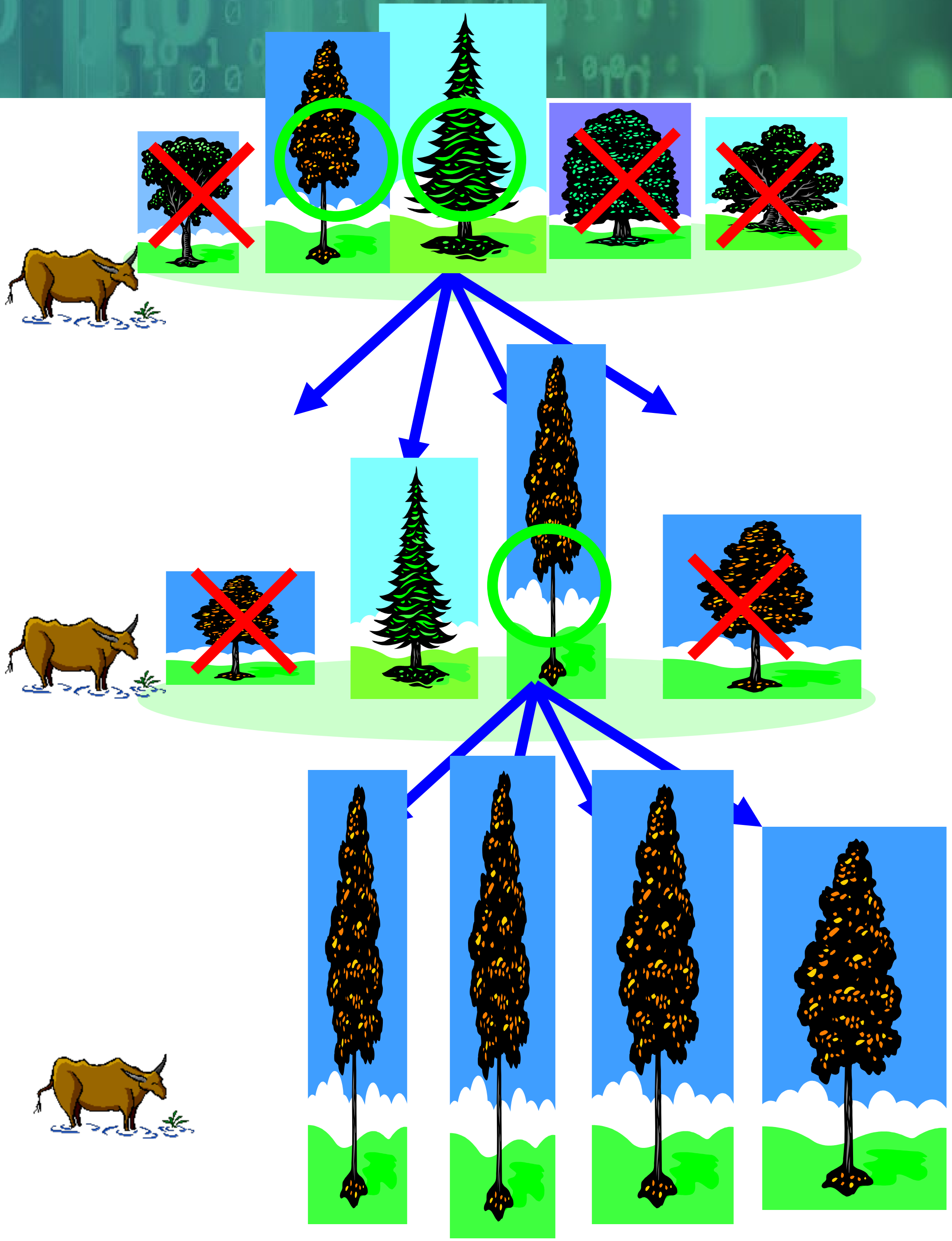
- ダーウィンの種の進化に必要な条件
  - 生物の個体は（遺伝子レベルで）変異する
  - 変異は遺伝する
  - 個体間に生存競争がある
    - 適者生存，自然淘汰

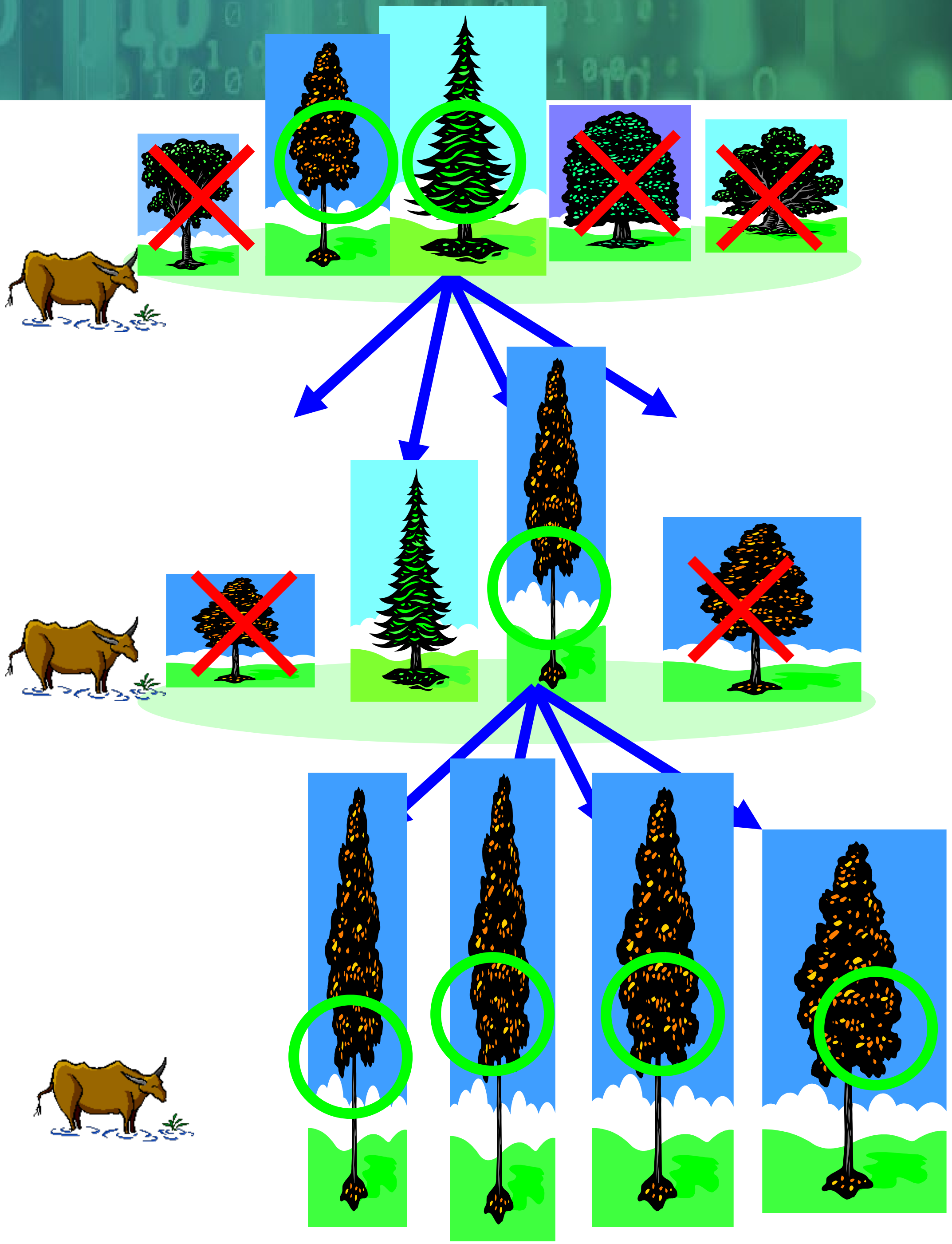






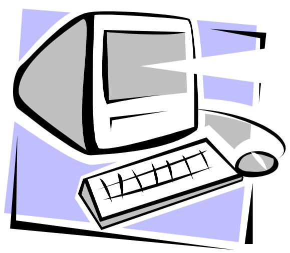




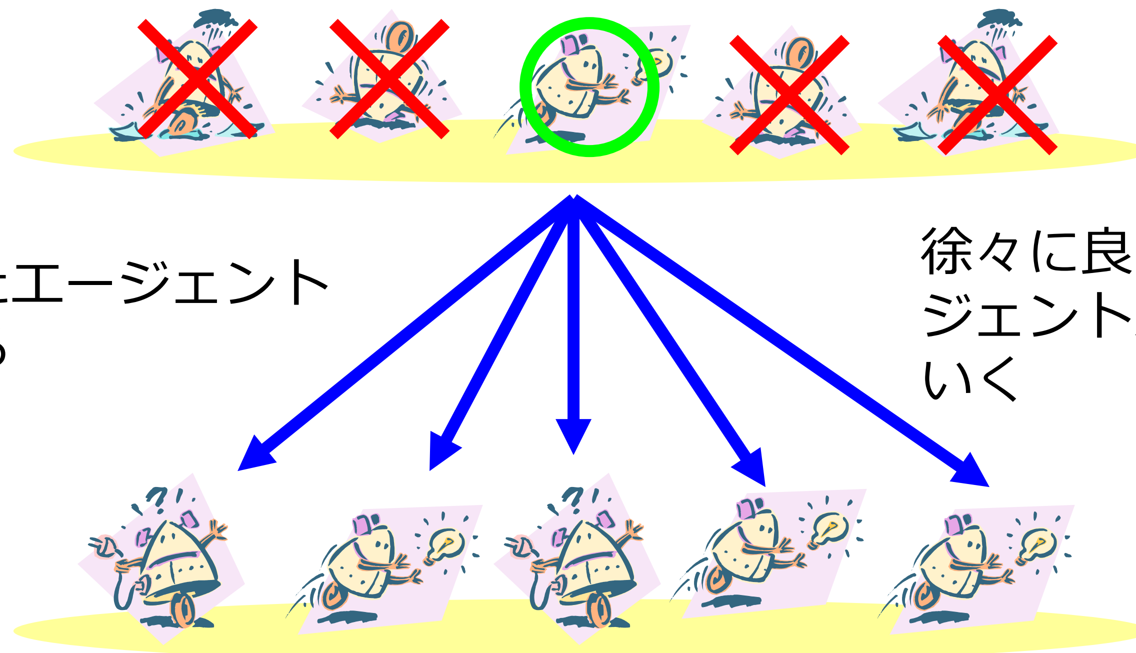


- 進化を問題解決に用いるときに必要な条件

（１）解（エージェント）は変異する.
（２）変異は複製される.
（３）解（エージェント）に対して評価が行われる.



よくできたエージェントが生き残る



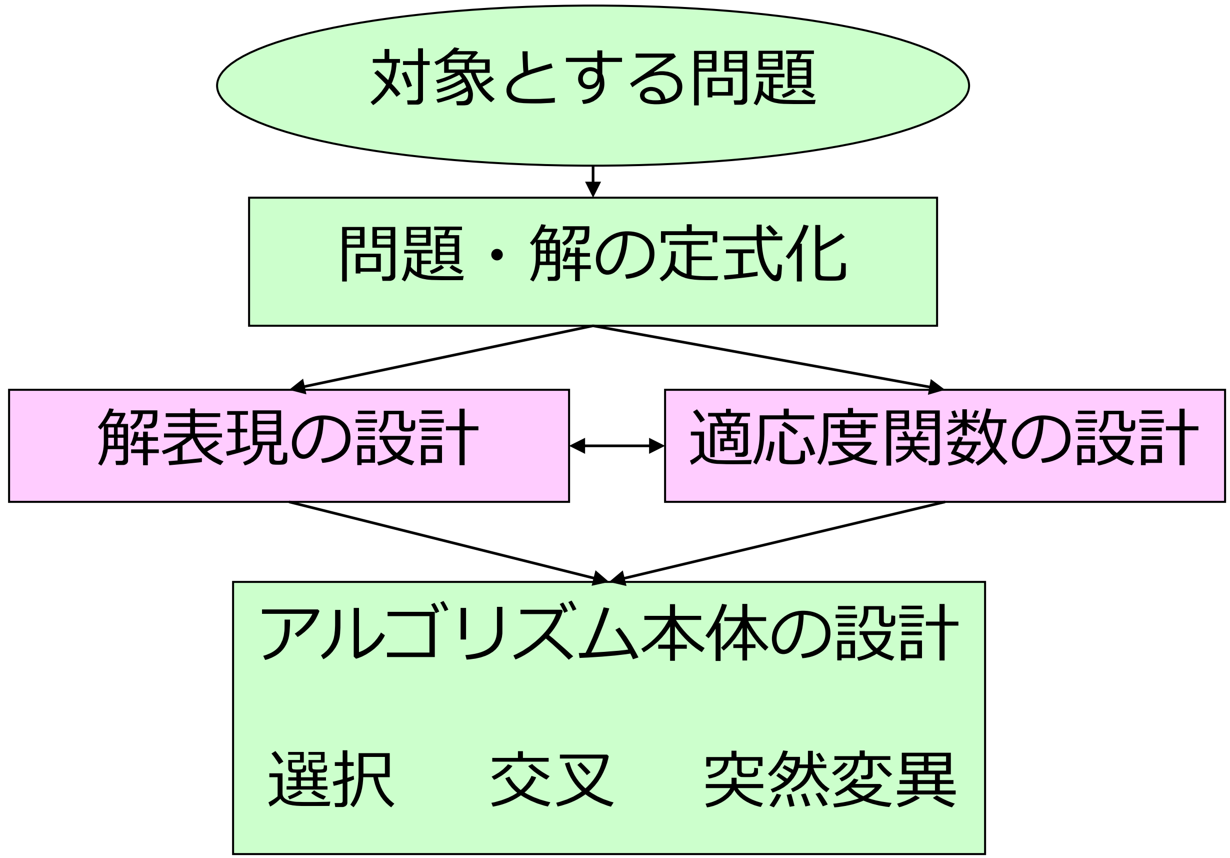
徐々に良いエージェントができていく

遺伝的アルゴリズム

- 1960年から1970年
  - John H. Hollandと彼の同僚やミシガン大学の学生等によって，自然界の適応過程を説明し，生物の進化のメカニズムを模擬する人工モデルとして提唱されてきた手法

- 原著
  - John Holland: Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence, University of Michigan Press(1975), The MIT Press(1992).
- 訳本
  - ジョン･ホランド著：遺伝的アルゴリズムにおける理論　－　自然・人工システムにおける適応　－，森北出版株式会社(1999).

遺伝的アルゴリズムによる問題解決



適用

　エージェントデザイン，組み合わせ最適化問題，システム設計等

遺伝的アルゴリズムの適応度

- 適応度
  - 対象となる遺伝子のよさを表す尺度
    - 通常大きい方が良い
    - $F_1<F_2,\ F_2<F_3 \rightarrow F_1<F_3$

- 遺伝的アルゴリズムの目的
  - 適応度の最大化

  - 適応度を設計して評価を行うということは，
    適応度さえ設計できれば様々な問題に適用できる．

単純遺伝的アルゴリズム（SGA）の流れ

ランダムな初期集団 $M(0)$ の生成

$M(t)$ 内の各個体 I を問題空間に適用し，適応度 F_i を計算

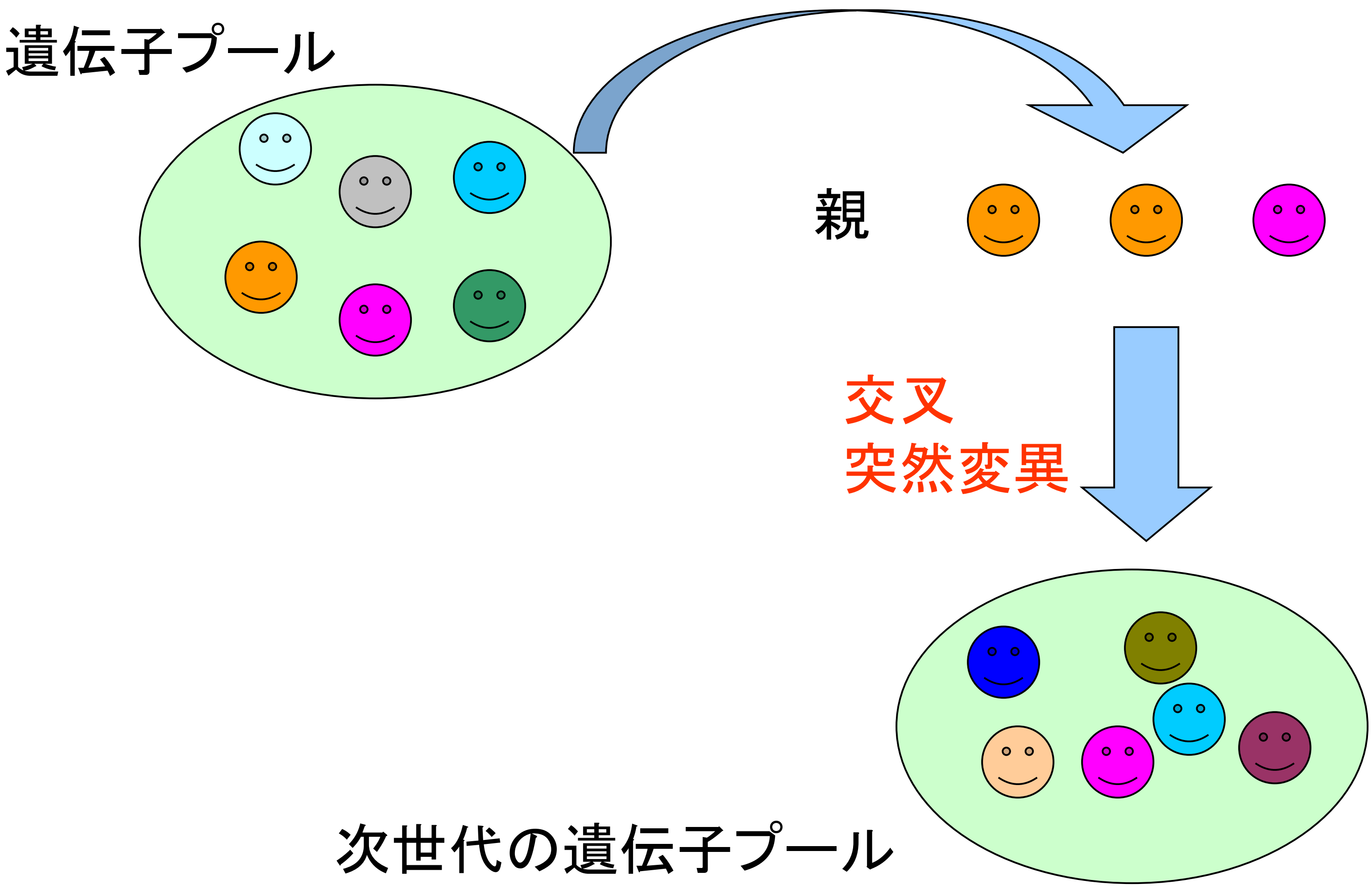
F_i に基づく確率 P_i に従って幾つかの個体を親として選択

選ばれた親に対して遺伝的操作を加えて次世代集団 $M(t+1)$ を生成

終了世代まで繰り返す

選択・遺伝子操作と世代交代

- 適応度に基づく選択



選択

- 選択
  - 適応度に従ってプールから個体を選択する操作
- ルーレット選択
  - 各個体の適応度に比例した確率で選択する.
  - 集団サイズだけ繰り返し，もとの集団サイズと同じ数が選択される（重複あり）

適応度 F_i の遺伝子の選択確率

$$P^i = F^i / \sum_{j=1}^{N} F^j$$

  - 長所：実装が簡単
  - 短所：スケールの問題がある
- エリート選択
  - 必ず集団内の最良個体を次世代へ残す選択法

スケーリング

- 適応度の評価が不適当であると、選択の際に不都合が生じることがあるため、適合度を適当なものに変換

- 線形スケーリング
  - スケーリング前後で平均適応度が変化しないように設定

$$F'^i = \alpha F^i + \beta$$

ただし,

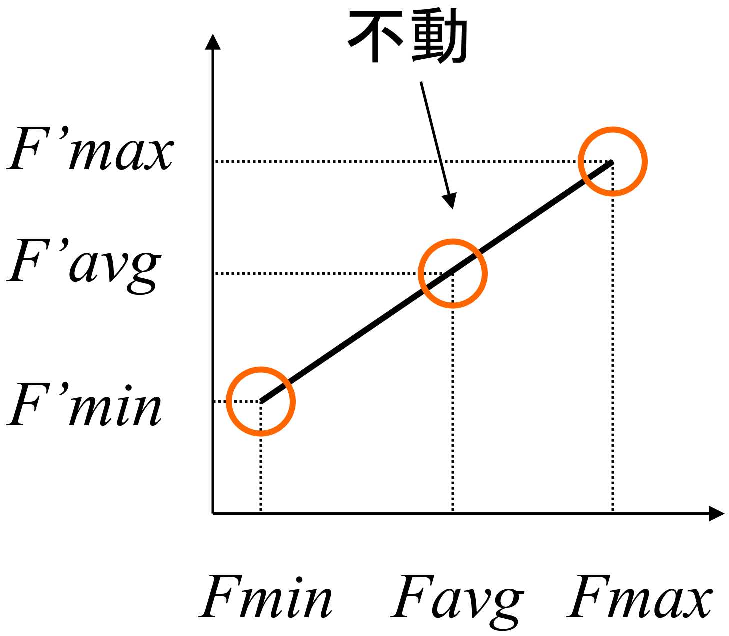
$$F'_{avg} = F_{avg}$$

$$F'_{\max} = C_{multi} \cdot F_{avg}$$

とするために,

$$\alpha = \frac{(c_{multi} - 1)F_{avg}}{F_{\max} - F_{avg}}$$

$$\beta = \frac{(F_{\max} - c_{multi} \cdot F_{avg}) \cdot F_{avg}}{F_{\max} - F_{avg}}$$



他の選択の種類（1）

- ランク選択
  - 適応度のスケールや集団内の分布の状況に依存せずに，集団内での適応度の順位に基づいて選択確率を決める
  - 適応度順にソートしたあと，以下の式によって選択確率を計算

$$P^i = \frac{1}{N}\left(\eta^+ - (\eta^+ - \eta^-)\cdot\frac{i-1}{N-1}\right)$$

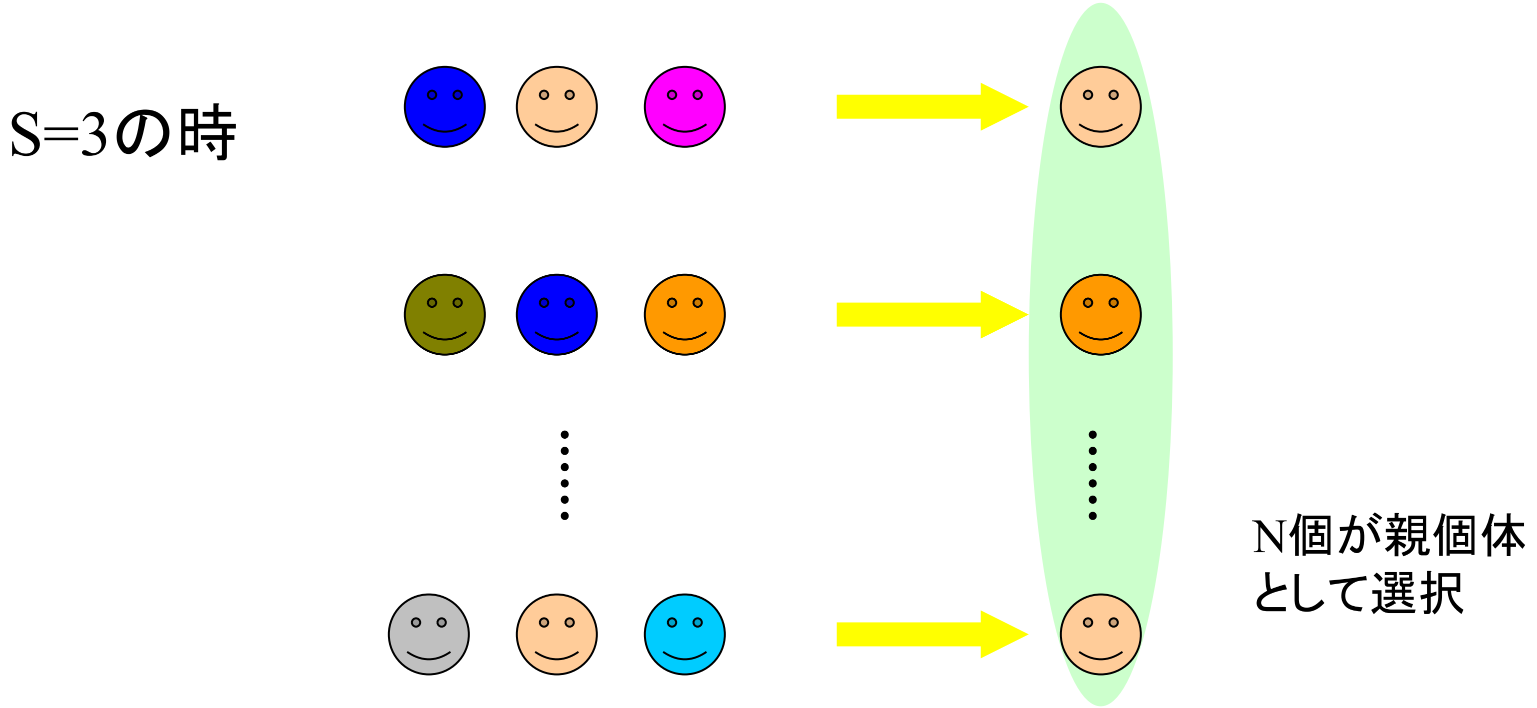
η^+ / N : 適応度が最良の個体の選択確率

η / N : 適応度が最悪の個体の選択確率

  - 利点：スケールによらない．
  - 欠点：固体のソート等に計算コストがかかる．

他の選択の種類（2）

- トーナメント選択
  - ランダムにs個の個体を選び，その中で一番よい個体を選ぶという操作をN回繰り返して選択を行う

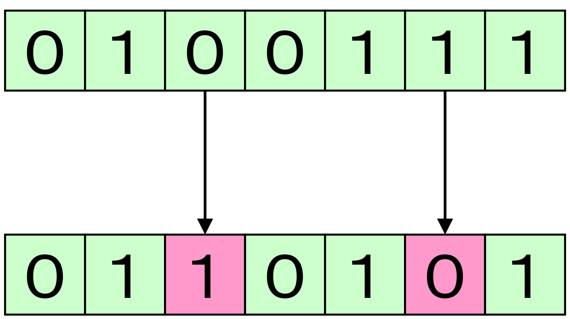


  - 利点スケールによらない．

突然変異

- 突然変異
  - 遺伝型にランダムな変異を加えることで、もとの遺伝子より僅かに違う遺伝子を作る操作

- Binary Cording の一般的な突然変異

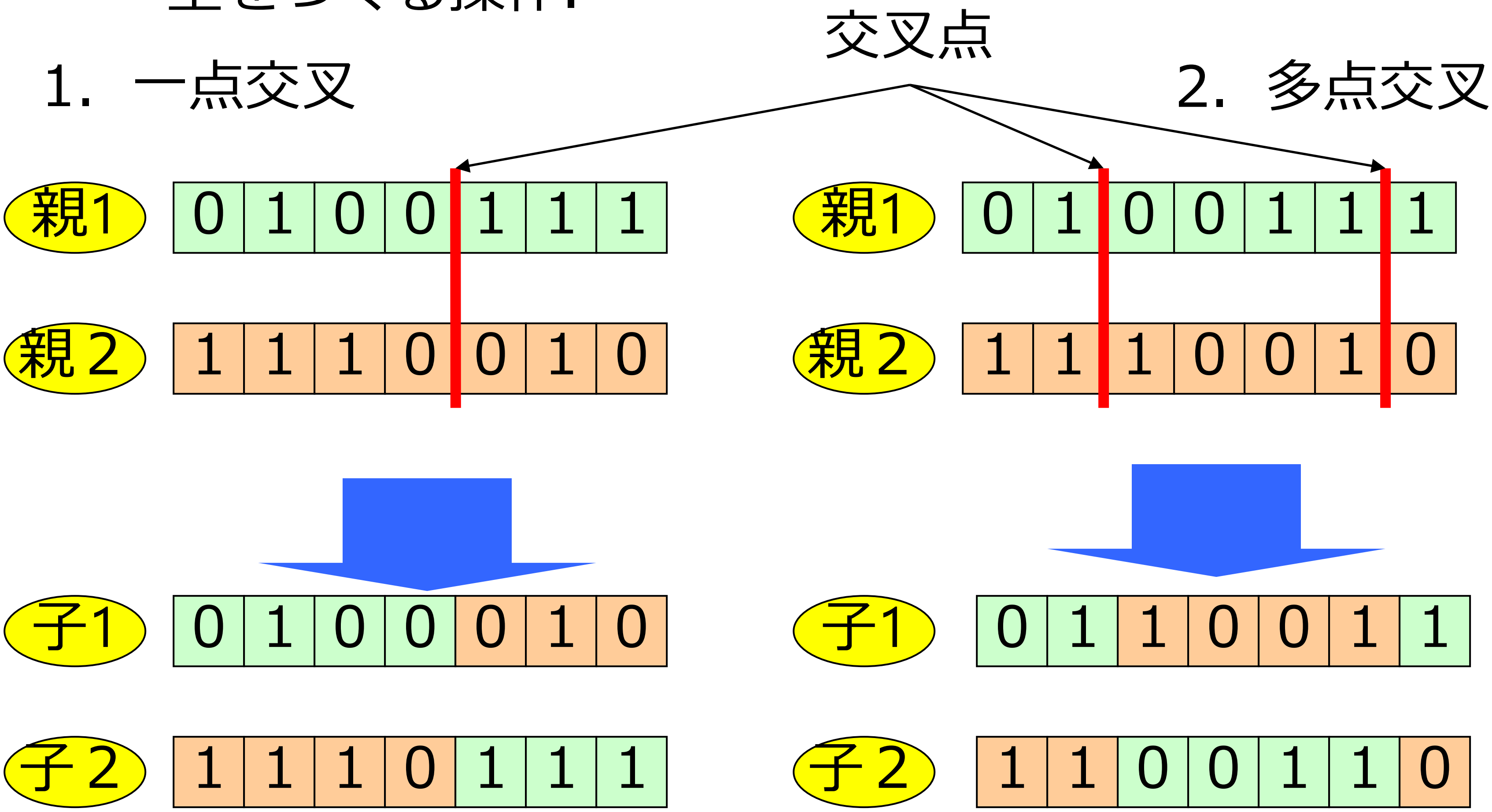


ある確率*Pm*で選ばれたビットを反転

  - 突然変異により近傍探索が進む
  - 遺伝子型で僅かな違いであっても，表現型では大きな違いになってしまうこともあるので注意が必要．

交叉（1）

- 交叉
  - 2つの遺伝型から，特徴を受け継いだ新たな遺伝型をつくる操作.



交叉（2）

3. 一様交叉

親1	0	1	0	0	1	1	1
親2	1	1	1	0	0	1	0
	0	0	1	0	1	1	0
子1	0	1	1	0	0	1	1
子2	1	1	0	0	1	1	0

ランダムに生成されたマスク

単純遺伝的アルゴリズム（SGA）の流れ

ランダムな初期集団$M(0)$の生成

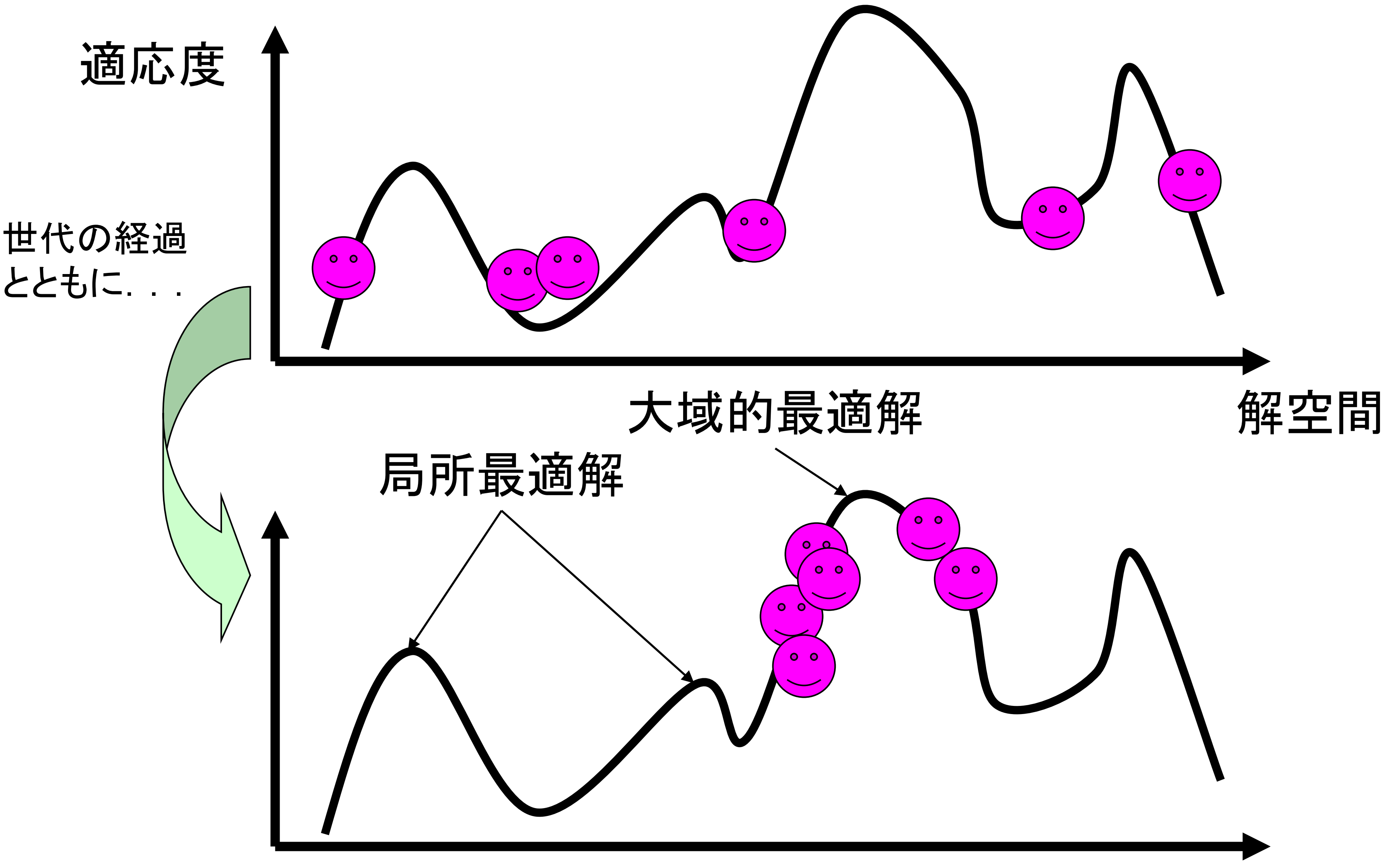
$M(t)$内の各個体iを問題空間に適用し，適応度F^iを計算

Fiに基づく確率Piに従って幾つかの個体を親として選択

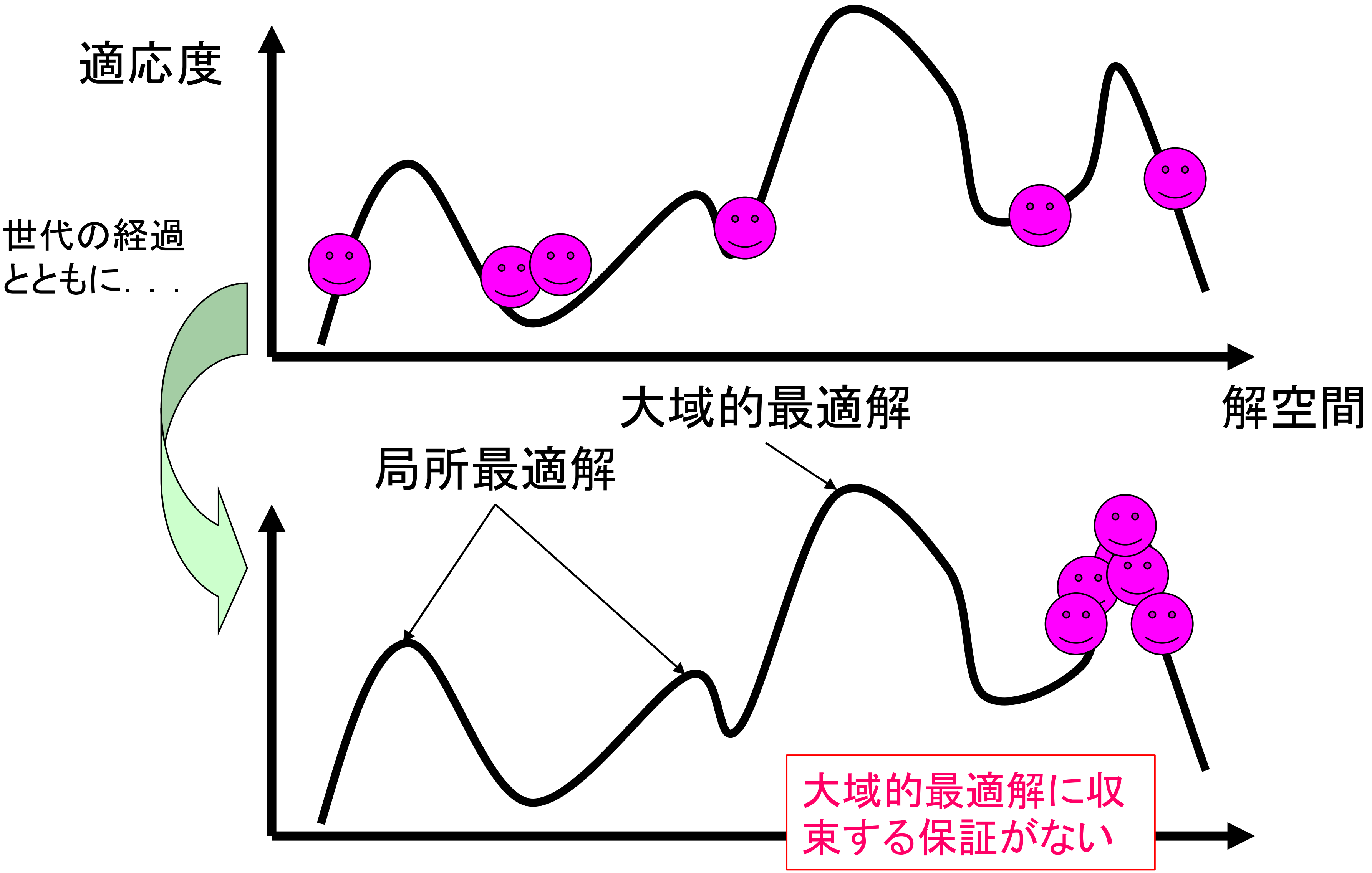
選ばれた親に対して遺伝的操作を加えて次世代集団$M(t+1)$を生成

終了世代まで繰り返す

遺伝子の集団的挙動（1）

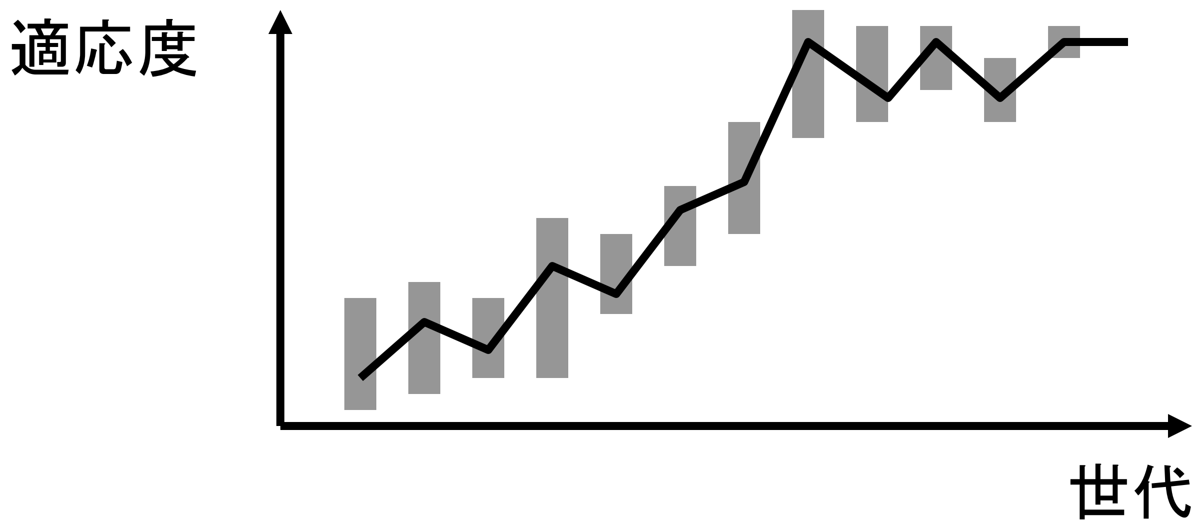


遺伝子の集団的挙動（1）



進化と適応度の変化

- 以上のアルゴリズムにより，進化過程が進む
  - 進化の初期はランダムな解から出発
  - 適応度が高い個体が生き残るので，徐々に平均適応度が上昇
  - 個体の改良が困難になり，ついには収束

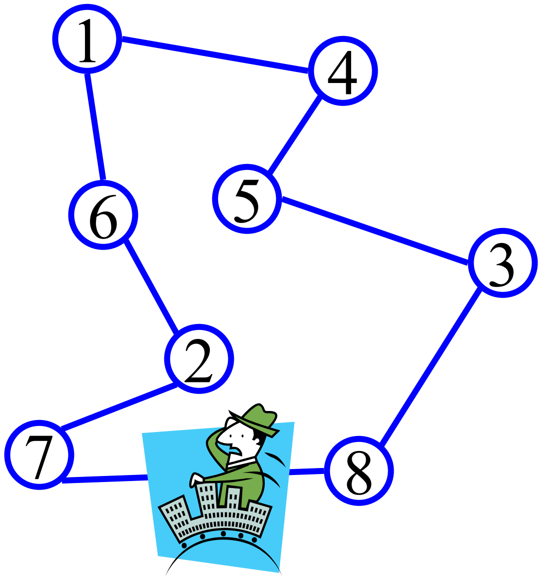


遺伝的アルゴリズム設計のポイント

- 遺伝子型（解表現）をどう設計するか
- 表現型をどう評価するか
- 遺伝子型の変化をどう設計するか

遺伝的アルゴリズムの応用例（１）

- 巡回セールスマン問題（TSP）



遺伝子表現:
都市の巡回順
例: 1 6 2 7 8 3 5 4

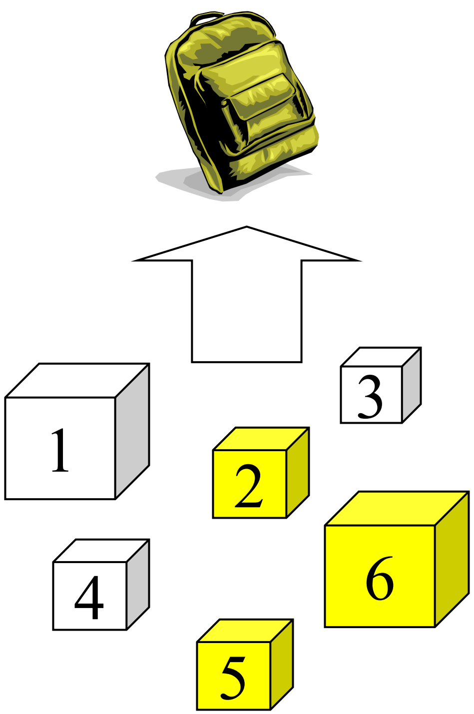
適応度:
総巡回路長

すべての都市を一巡する総移動距離
が最小の巡回路を求める問題

都市数がnのとき，全組み合わせは
(n-1)!/2

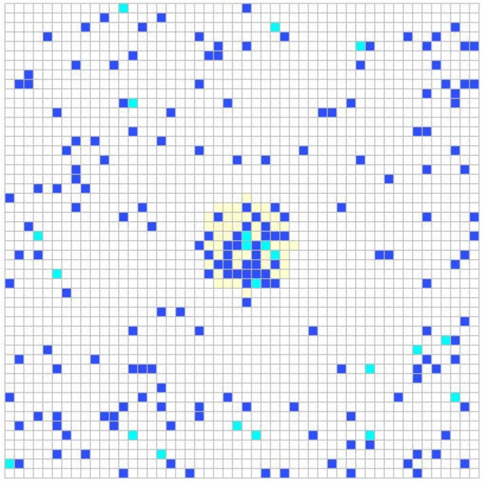
遺伝的アルゴリズムの応用例（２）

- ナップザック問題
  - 複数の物体が与えられたとき，重さがある範囲内でいくつかの物体を選択し，その価値の合計が最大になるような組み合わせを求める問題
  - 遺伝子表現：
    - 物体iが選択されているとき１
    - されていないとき０
    - 例：010011
      - 荷物1, 4, 5 は選択
      - 荷物0, 2, 3 は選択せず
  - 適応度
    - 価値の合計



遺伝的アルゴリズムの応用例（３）

- 人工社会のエージェント設計



上下左右に砂糖の多い場所があるか？
0000 〜 1111

上下左右の移動
1 〜 4

0000	1
0001	1
0010	4
0011	2
0100	3

・・・

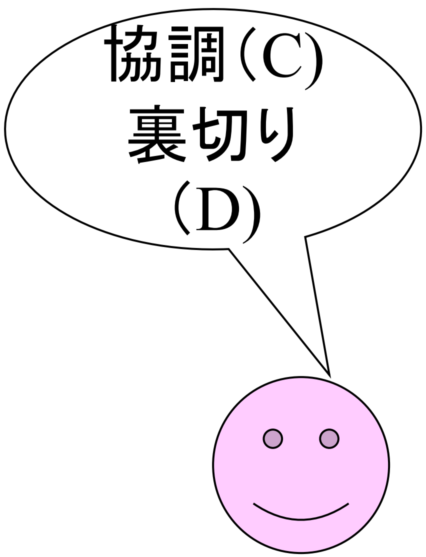
遺伝子表現

個々のエージェントは生き残るために砂糖を獲得しようと行動するが，砂糖を多く持っているエージェントが必要以上に砂糖を集めると，砂糖を手に入れられないエージェントが死滅してしまう.

果たしてどういう行動ルールがよいのか？

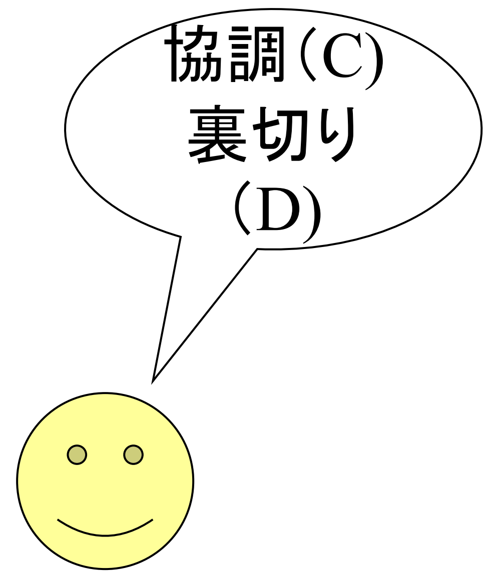
適応度：
生き残りエージェント数

繰り返し囚人のジレンマゲーム



得点票

	C	D
C	(3,3)	(1,4)
D	(4,1)	(2,2)



同時に選択する手を決める.

互いに相手がどのような手を取ろうとも，Dのほうが利得が高いので，Dを選ぶのが合理的. しかし，互いにDを選んで得られた結果(1,1)よりも，互いに協調を選んだ結果(3,3)のほうが良い結果になるという意味で，解決困難な問題のひとつ.

囚人のジレンマ1

2人の容疑者は隔離…相談は不可能

		容疑者2 黙秘	容疑者2 自白
容疑者1	黙秘	1年,1年	10年,3ヶ月
	自白	3ヶ月, 10年	8年,8年

右が容疑者1の刑期、
左が容疑者2の刑期

二人とも黙秘…二人とも懲役1年　一方が黙秘、一方が自白
二人とも自白…二人とも懲役8年　黙秘…懲役10年　自白…懲役3ヶ月

「囚人のジレンマ」の事例

割り勘、税収、環境問題、軍拡競争、etc.

囚人のジレンマ2

- 囚人のジレンマの利得行列(双行列表現)

プレイヤー2

$$
\begin{array}{ll}
 & \quad\;\;\mathbf{C} \qquad\quad \mathbf{D} \\
\text{協調}(\mathbf{C}\text{ooperate}):\mathbf{C} & \multirow{2}{*}{$\begin{bmatrix} 3,3 & 1,4 \\ 4,1 & 2,2 \end{bmatrix}$} \\
\text{裏切}(\mathbf{D}\text{efect}):\mathbf{D} &
\end{array}
$$

プレイヤー1

（協調＝黙秘、裏切＝自白）

右がプレイヤー1の利得
左がプレイヤー2の利得

囚人のジレンマ3

容疑者１の合理的な思考

もし相手が黙秘するならば，
自分は自白をするのが適当である．

もしそうすると，相手は黙秘
から自白に変えるであろう．

そのとき自分は黙秘に変えると刑期
が延びてしまうから，自白に留まる．

容疑者２の合理的な思考

もし相手が黙秘するならば，
自分は自白をするのが適当である．

もしそうすると，相手は黙秘
から自白に変えるであろう．

そのとき自分は黙秘に変えると刑期
が延びてしまうから，自白に留まる．

容疑者１の思考と容疑者２の思考

2人とも自白を選択…2人とも懲役８年

ナッシュ均衡とは

非ゼロ和ゲームの均衡解

- 自己拘束性の必要性
  - 均衡解に自己拘束性がなければ必ず少なくとも一人のプレイヤーはその均衡から離脱する動機を持つ. このときゲームの解自体は拘束力のある合意に　基づくものではないから, 実際のプレイがゲームの解と一致する保証はない
- ナッシュ均衡解

相手のナッシュ均衡戦略に対して自分の利得を最大にする戦略はナッシュ均衡戦略である

他のすべてのプレイヤーがナッシュ均衡点に従うと予想するとき, どのプレイヤーも自らそのナッシュ均衡点から離脱する動機を持たない

自己拘束性を持つため非ゼロ和ゲームにおける合理的行動の解となり得る

無限繰り返しゲームの定式化１

- 2人非ゼロ和ゲーム

１．プレイヤーの数は2人
2　 利得の和はゼロとは限らない
３．各プレイヤーのとりうる戦略の数は有限
４．ゲームは一回限り
５．各プレイヤーは相手の戦略に関して情報を持たない

⬇

2人無限繰り返しゲーム

4’. 同一のゲームを無限に繰り返しおこなう

プレイヤーは過去のプレイの結果に依存して行動

繰り返しゲーム

マルチエージェントシステム

複数のエージェントの連続的に繰り返される意思決定

どのような記述が適しているか？

「ある環境状況が繰り返し訪れる中での意思決定」と捉える

繰り返しゲーム…戦略形ゲームの繰り返しにおける意思決定

マルチエージェントシステム

＝ 終焉のない動的な環境

2人無限（有限）繰り返しゲーム

2人非ゼロ和ゲームを無限（有限）回繰り返しおこなう

無限繰り返しゲームの定式化２

1. 成分ゲーム

成分ゲーム : 戦略形ゲーム $G = (N, S, F)$

- プレイヤー集合 $N = \{1,2\}$
- 行動集合 $A = A_1 \times A_2$

  A_1 ,A_2 : プレイヤー1とプレイヤー2の行動の集合
- 利得関数 $f = f_1 \times f_2$

  f_1, f_2 : プレイヤー1とプレイヤー2の利得関数

2. 繰り返し回数

成分ゲームG をt = 0 から t = T まで繰り返す

T : 有限な自然数 ➡ 有限繰り返しゲーム G^T

T : 無限大 ➡ 無限繰り返しゲーム G^∞

無限繰り返しゲームの定式化 3

3. 戦略

- 成分ゲームの純戦略

  繰り返しゲームの **行動の決定**

  プレイヤー1とプレイヤー2の行動 $a_1 \in A_1, a_2 \in A_2$

- 繰り返しゲームの戦略

  過去のプレイの結果に依存して毎回成分ゲームの行動を決定する指針

  プレイヤー1とプレイヤー2の戦略 $s_1 \in S_1, s_2 \in S_2$

ex.) 囚人のジレンマ　行動 : **C** または **D**

戦略 : しっぺ返しの戦略

- 行動の列

  プレイヤーの戦略の組 $\mathbf{s} = (s_1, s_2)$ に対して、行動の列が定まる

$$\mathbf{a}(\mathbf{s}) = (\mathbf{a}^1, \ldots, \mathbf{a}^T),\ \ \mathbf{a}^t = (a_1^t, a_2^t) \quad t = 1,2,\ldots$$

4. ゲームの履歴

成分ゲームGのk 回目のプレイの結果

繰り返しゲームの特徴的な要素

- その回のプレイヤーの利得の組　$\mathbf{x} = (x_1, x_2)$
- その回の選択された純戦略の組　$\mathbf{a} = (a_1, a_2)$

これらを用いてk 回目のプレイの結果をo^k　と表現すると

1回目からt 回目までの結果の系列を以下のように表す

$$\mathbf{h}(t) = (o^1, o^2, \ldots, o^t)$$

この結果の系列を**ゲームの履歴**という

無限繰り返しゲームの定式化 5

5. 利得

t=0 における将来にわたって得られる利得についての期待値

繰り返し期間が長い場合 ➡ 一般的に近い将来に得られる利得に比べて遠い将来の利得は割り引いて考える

割引因子 δ の導入 $(0 < \delta < 1)$

t 回目のプレイヤー1の利得の現在価値（割引利得）:

$$\delta^t f_1(a_1^t, a_2^t)$$

プレイヤー1の割引利得の総和（割引利得和）:

$$F_1(s_1, s_2) = \sum_{t=1}^{\infty} \delta^t f_1(a_1^t, a_2^t)$$

合理的なプレイヤーは**割引利得和**の最大化を目指す

無限繰り返しゲームの定式化 6

6. 共通認識

プレイヤーはこれらの要素に関して共通認識がある

- 成分ゲーム G
- 繰り返し回数 T
- ゲームの履歴 **h**(t)

7. ゲームの前提条件

合理的なプレイヤー

1) 利得の最大化を目指す ➡ 1‘) **割引利得和**の最大化を目指す

2) 相手の行動を可能な限り推論

自分の選択した戦略に対して相手が割引利得和を最大にするような最適な戦略を選択するとして戦略を選択

再掲：囚人のジレンマ

- 囚人のジレンマの利得行列

プレイヤー2

プレイヤー1

$$\begin{array}{cc} & \begin{array}{cc}\mathbf{C} & \mathbf{D}\end{array} \\ \begin{array}{r}\text{協調(}\mathbf{C}\text{ooperate)}:\mathbf{C} \\ \text{裏切(}\mathbf{D}\text{efect)}:\mathbf{D}\end{array} & \begin{bmatrix} 3,3 & 1,4 \\ 4,1 & 2,2 \end{bmatrix}\end{array}$$

（協調＝黙秘、裏切＝自白）

左がプレイヤー1の利得
右がプレイヤー2の利得

ナッシュ均衡点

プレイヤーの合理的な意思決定の結果

無限繰り返し囚人のジレンマ 15

無限に繰り返される囚人のジレンマ

ゲームの環境として長期にわたる相互関係が保障

＋

十分に大きい割引因子δ

プレイヤーが将来の利得を高く評価 →長期の相互関係と認識

一回ゲーム… (D,D)の系列を実現する戦略の組のみがナッシュ均衡

繰り返しゲーム… (C,C)の系列を実現する戦略の組がナッシュ均衡に含まれる

ゲームの無限繰り返しによる
協調の可能性の発生

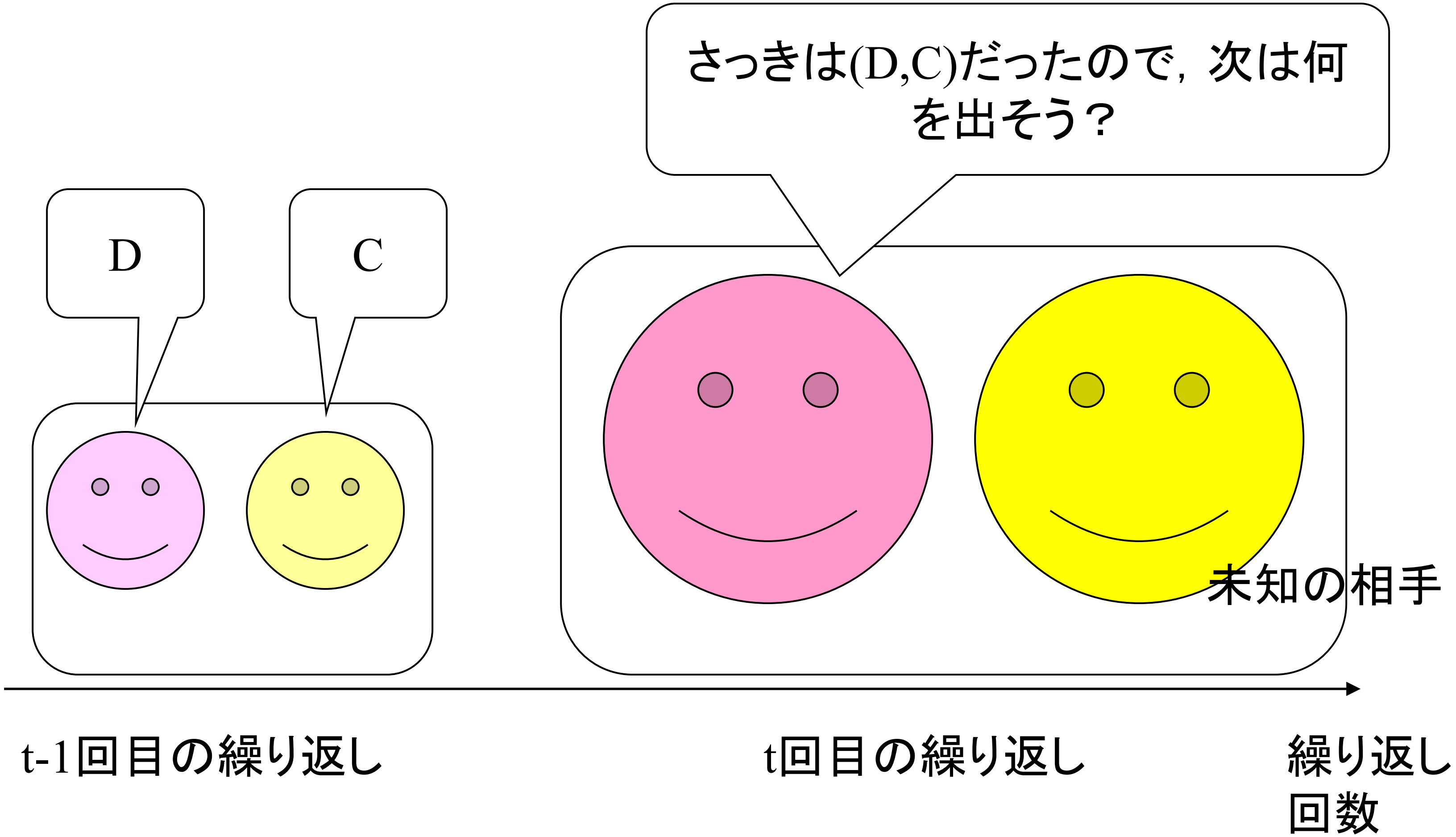
繰り返し囚人のジレンマにおけるエージェント設計

- 繰り返し囚人のジレンマゲーム
  - 囚人のジレンマゲームを繰り返し（ここでは不定回数）行うゲーム
  - 目的
    - そのときの平均利得を大きくするのが

- エージェント設計
  - エージェントは，前回までの相手と自分のとった手を基に，次の自分の手を決める
  - このとき，どういう意思決定をすると良いか？

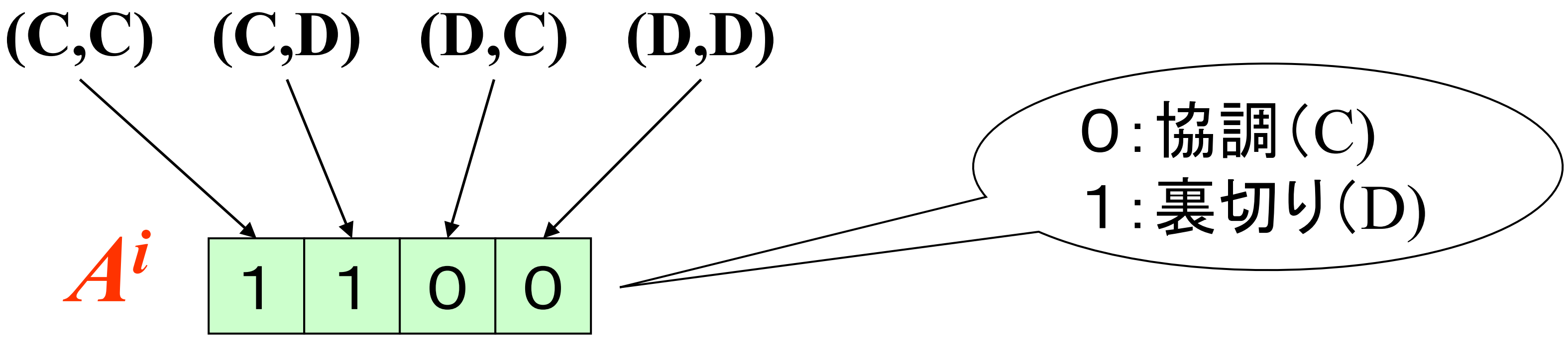
繰り返し囚人のジレンマ

- 繰り返し囚人のジレンマにおける
  記憶長2の意思決定システムの設計



意思決定システムの解表現の設計

- 記憶長2の場合，考えうる状況(C,C),(C,D),(D,C),(D,D)の4通りに対して何を出すかが表現できればよい．



Binary Cording（０，１表現）による解表現

記憶長がnの時，Binary cordingによる表現は２のn乗になる．

※状態遷移図表現等，他の解表現も考えられる．

遺伝子型と表現型

遺伝子型(Genotype)

A^i

1	1	0	0

一意に
写像

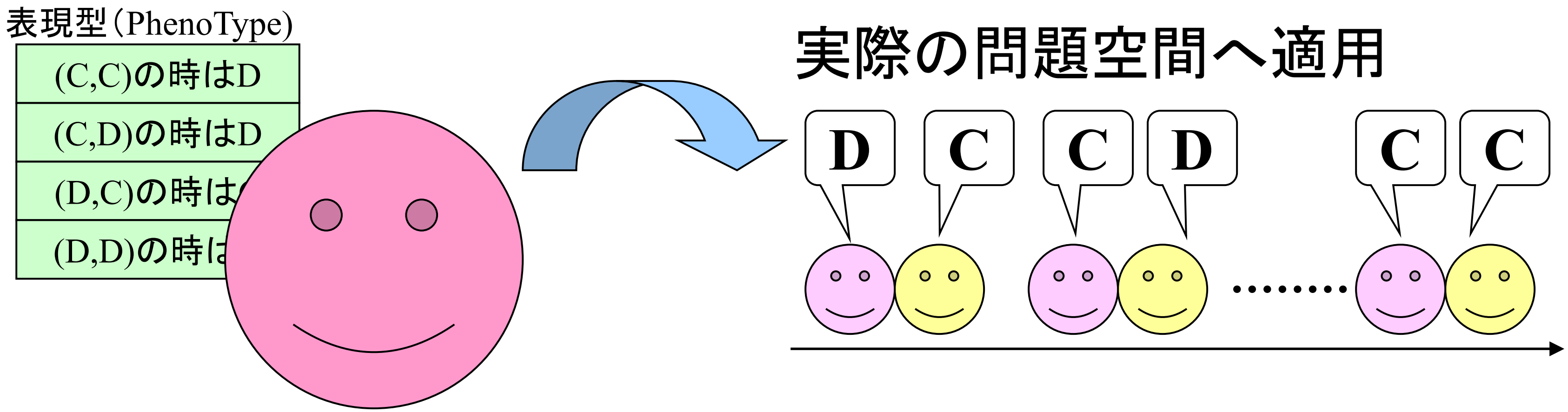
表現型(Phenotype)

(C,C)の時はD
(C,D)の時はD
(D,C)の時はC
(D,D)の時はC

遺伝操作
(Genetic Operators)
の操作対象

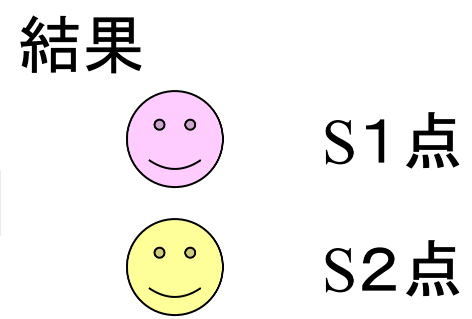
問題空間において,
評価される対象

解評価



結果から適応度(Fitness)を決定

$F^i = f(A^i)$



$F^1 = S1$

$F^1 = S1 - S2$

$F^1 = S1/S2$

etc….

しっぺ返し戦略(tit-for-tat) の発見

- 前回のゲームで相手の出した手を次のゲームで出す
  - (自分, 相手) ： CC ⇒ 次回C
  - (自分, 相手) ： CD ⇒ 次回D
  - (自分, 相手) ： DC ⇒ 次回C
  - (自分, 相手) ： DD ⇒ 次回D
    - 通常、初回はCを出す
    - その後、自分が何を出したか、どのような組み合わせになったかは関係ない

繰り返し囚人のジレンマ コンテスト 1

繰り返し囚人のジレンマの分析

- ゲーム全体を俯瞰する立場からの考察

ex.) 複数のナッシュ均衡点、フォークの定理

- プレイヤー個人の立場からの考察

有限繰り返し囚人のジレンマ

ナッシュ均衡戦略：all-D…高い利得を獲得できるか？

➡ お互いにDを選択すれば、お互いCよりも低い利得

実際に繰り返し囚人のジレンマをおこなう場合、
どのような戦略を選択すればよいのだろうか？

繰り返し囚人のジレンマ コンテスト [Axelrod 1984]

複数の戦略プログラムの総当たり対戦

繰り返し囚人のジレンマコンテスト2

第1回コンテストの概要

プレイヤー	心理学、経済学、政治学、数学、社会学の分野に属する14名に作成されたプログラム ＋ ランダムプログラム
ルール	総当り対戦…各対戦は200回繰り返しを5回おこなう 評価…対戦で得られた利得の合計 コンテストで用いられた利得行列 $\begin{array}{cc} & \begin{array}{cc} C & D \end{array} \\ \begin{array}{c} C \\ D \end{array} & \begin{bmatrix} 3,3 & 1,4 \\ 4,1 & 2,2 \end{bmatrix} \end{array}$
結果	優勝… 上位を占めた戦略の特徴 自分から裏切らない＝上品さ(nice) 相手が裏切っても再び協調し合える＝心の広さ（forgiveness） キングメーカーの存在

繰り返し囚人のジレンマコンテスト3

第2回コンテストの概要

プレイヤー	前回の分野 +コンピュータサイエンス、物理学等の分野に属する62名に作成されたプログラム +ランダム プログラム
ルール	前回のルール + 繰り返し回数の確率的変動
結果	第1回コンテストの結果を踏まえた参加プログラムの２つの傾向 1．上品で心が広いプログラム　（しっぺ返しの戦略の踏襲） 2. 1のようなプログラムから搾取を狙うプログラム → 1 同士では協調、2 同士で裏切り合い 優勝… しっぺ返しの戦略 上位を占めた戦略の特徴 上品で心が広い… 傾向1 → 前回と同様 非協調的な相手（ex. all-D）には裏切り

- しっぺ返しの戦略の後の有効な戦略
  - パブロフ
    上手くいかなかったら次回の手を変更
    - 上手くいったので変更なしの場合
      - (自分, 相手) ： CC ⇒ 次回C、(自分, 相手) ： DC ⇒ 次回D
    - 上手くいかなかったら変更ありの場合
      - (自分, 相手) ： DD ⇒ 次回C、(自分, 相手) ： CD ⇒ 次回D
  - シグナル
    初期に決まった手を出した相手とだけ協調
      - 例）最初の5回　DDCDC を出した相手とCを出し合う

「囚人のジレンマコンテスト」への批判

- シミュレーション全般に対する注意点
  - 過度な擬人化
    - エージェントは単なるプログラム
      - 実装していない機能に関しては語れない
  - 試行数の不足
    - 10や20の設定では少な過ぎる
      - 「ある特殊な条件で成り立つ」可能性を示しているだけ
      - 統計的な議論をするべき
  - 拡大解釈
    - シミュレーション： 現実→モデルという縮約プロセス
      - 現実の多くの要素を捨象しているので、シミュレーションの結果が必ず現実の現象に還元できるわけではない

遺伝的アルゴリズムの遺伝子型

- 遺伝的アルゴリズムの遺伝子型は一般に特別な構造をもたない

- エージェントの設計においては，遺伝子型と意思決定システムの対応を考える必要があるが，どのような対応がよいのかわからない

- 構造をもつ遺伝子型（いわゆるプログラム等）による遺伝的アルゴリズム