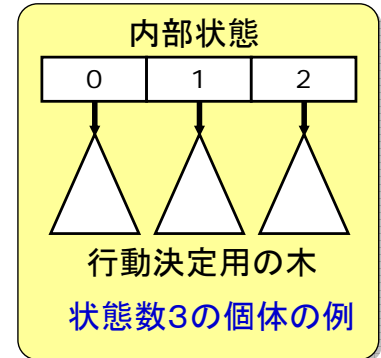


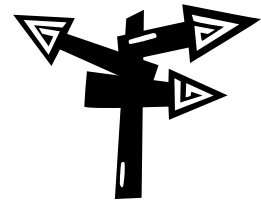
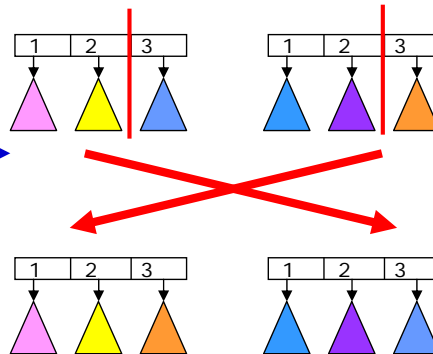
適応型GPオートマトン AGPA

- あらかじめ様々な状況を想定する必要があるなどの理由により、エージェントの行動制御プログラムを手で作ることが困難な場合、進化計算法の一つである遺伝的プログラミング(GP)が用いられることがある。
- しかしながら、通常のGPでは同一入力に対して同一の行動出力しか得ることができない。
- これを改善するため、エージェントに内部状態を設定し、各状態毎に行動決定用の木をもたせた GP-オートマトン(GPA) が Ashlock によって提案された。



- 状態数はあらかじめ指定する必要がある。

- GPAにおける交叉:

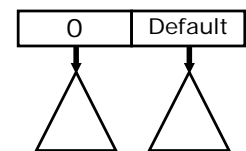


1

適応型GPオートマトン AGPA

- GPAでは、エージェントの状態数をあらかじめ指定する必要がある。問題の複雑さは事前に解らないことが多いため不便であった。
- そこで、与えられた問題の複雑さに応じてエージェントの状態数を世代交代中に増減させて調整することができる AGPA (Adaptive-GPA) を提案する。

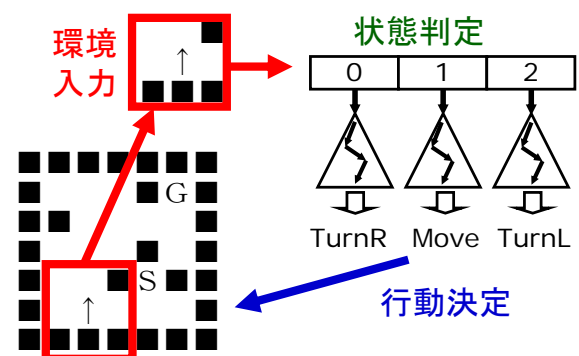
- 初期個体は初期状態と未定義の状態を処理する木から構成される(右図)。



- 終端記号は状態番号に対する操作と行動のセットである。

- 世代交代の際に状態数の増減を行う。
- 状態番号に対する操作は次のいずれか。
 - Inc: 状態番号を1増やす。
 - Dec: 状態番号を1減らす。
 - Set i: 状態番号を i にする。
 - Stay: 状態番号を維持する。

- AGPA を右に示すような迷路探索問題に適用して有効性を確認することができた。



2