

探索アルゴリズム(新保仁)

実時間探索. 自律エージェントの基本アーキテクチャの探求は、石田亨が京都大学赴任前より継続して取り組んだ課題である。石田が注目した手法の一つに、“実時間探索”があった。

A*をはじめとする従来の AI 探索法が、プランニング（目標達成までの完全な動作系列の発見）のみを取り扱い、動作の実行については一切考慮しないのに対し、実時間探索では、短期的な先読みに基づき次動作を決定し即座にその動作を実行する。そして、先読み・動作実行という一連の手順が、目標が達成されるまで繰り返される。結果として、問題全体の完全なモデルがあらかじめ既知ではない場合や、問題解決過程で生じた状況の変化に対応する柔軟性が生まれる。

石田が実時間探索に注目したもう一つの理由は、それが備える学習能力にあった。実時間探索では、短期的な先読みに基づき動作が決定されるため、実行された動作系列が最適という保証はない。しかしながら、ある種の実時間探索アルゴリズムには、同一あるいは類似の問題に対し繰り返し適用すると、最終的に最適な動作系列を発見する、という性質がある。この性質は、アルゴリズムが試行錯誤を通じて、正しい動作を“学習”する能力を持つことを示している。

自律的に学習するエージェントとは、まさに石田の望むところであった。研究室発足初年度には、水野智文とともに、実時間探索の学習性能評価を行なった。この研究は、後に研究室最初の学術論文として人工知能学会誌に採録された。水野との研究によって明らかになった実時間探索の問題点への解法として、さらに石田は新保仁と**重みつき実時間探索**および**上界値を用いた実時間探索**の提案を行なった。これらの手法はいずれも、エージェントアーキテクチャとしてより望ましい手法は何か、という観点から実時間探索の有効性を探ったものである。

探索アルゴリズムに基づく配列整列問題の解法. 配列整列問題は複数の遺伝子配列間の近さを測るタスクであり、遺伝機能の予測などに用いられる。7 配列程度であれば従来の A* 探索で対応できるが、8 配列以上に対してこの方法を用いると、メモリ不足に落ちいってしまい解くことができない。この難問の解決を研究テーマとして突き付けられたのが三浦輝久である。

探索においてメモリ不足緩和の常套手段は、反復深化である。ただ、問題空間が束 (lattice) をなす配列整列問題においては、問題空間内の同一の節点を通る数多くの経路が存在するため、出発節点からの経路を順次たどる反復深化とは恐ろしく相性が悪い。事実、反復深化 A* を用いて現実的な時間で解くことができる問題の規模は、3 配列程度にすぎない。

このため過去、限られた大きさのキャッシュを反復深化と併用する、いわば A* と反復深化 A* の中間的な手法が各種提案されてきた。これらはいずれも基本的に、やみくもに訪問節点を記憶しキャッシュが一杯になった時点で反復深化に変更する手法であった。

結果として、出発節点に近い節点ばかりが記憶されることになるが、効率の観点からは、より多く訪問される節点こそが記憶されるべきである。三浦は、この観察に基づき、**確率的節点記憶方式**を提案した。確率的節点記憶方式では、反復深化の過程で節点を訪問する都度コインを投げ、その節点をキャッシュに記憶するか否かを判断する。この手法により、節点が記憶される確率はその節点の実際の訪問回数を反映したものになり、結果として従来法に比べ 3 倍の高速化につながった。その後、吉住貴幸と三浦は、配列整列問題の持つ問題空間の性質（分岐率が大きいため、A* の生成節点数が展開節点数に対してはるかに大きくなる）に着目し、A* 探索の節点展開時にすべての子節点を生成するのではなく、有望な節点のみを順次展開する**段階的節点展開方式**を提案し、従来解くことができなかつた 8 配列の整列に成功した。彼らが着目したのは、束をなす問題空間に共通する性質であり、提案手法の適用範囲は極めて広い。