

メタヒューリスティクスに対する  
遺伝的プログラミングによる  
創発的パラメータ調整則の自動設計

2015 年度

金 政 実

# 主 論 文 要 旨

報告番号	甲 乙 第 号	氏 名	金 政 実
<p>主 論 文 題 目 :</p> <p>メタヒューリスティクスに対する遺伝的プログラミングによる創発的パラメータ調整則の自動設計</p>			
<p>(内容の要旨)</p> <p>最適化問題の計算手法として、大域的探索性能などの高機能を有するメタヒューリスティクスと称する発見的探索手法が数多く提案され、工学的な実問題に盛んに応用されるようになったが、その際に課題となることは、解きたい問題に適した手法の選択と、そのアルゴリズムの更新則中のパラメータの設定である。これは、メタヒューリスティクスが数理的な根拠に基づく手法ではなく、それらの最初の開発者と数多くの研究者による改良が、ベンチマーク問題を用いた計算機実験による経験や試行錯誤によってなされてきたことに大きく起因する。また、これらの改良手法の開発においては、実問題を解くときに許されないような過剰なファンクションコールによって評価が行われることが多く、計算コストが非常に大きい目的関数を有する実問題に適用したときに、かえって効率が悪化することがあり、ユーザの計算機環境に合わせたメタヒューリスティクスの効率的設計が望まれる。</p> <p>そこで本論文では、メタヒューリスティクスの更新則中のパラメータについて、探索状況に応じて動的に調整する調整則を、解きたい問題に適応して自動的に設計する方法論を提唱する。そのうえで、それらがアルゴリズムに与える影響を考察しつつ、その設計法として遺伝的プログラミング(以降 GP と略称)を用いる手法、および、その設計計算から創発される新しいメタヒューリスティクスのアルゴリズムを数種類提案する。すなわち、メタヒューリスティクスのアルゴリズムに対して、数多くの研究者やユーザによる開発と改良の試行錯誤過程を、メタ最適化問題として記述し、このメタ最適化問題を解くことを GP を用いることで自動化するという新しい方法論の提唱である。本論文は以下の構成となっている。</p> <p>第1章は、メタヒューリスティクスの歴史とその課題、および本論文の目的・意義・構成について概説し、第2章において、メタヒューリスティクスとして Particle Swarm Optimization (PSO: 粒子群最適化)、Evolution Strategy (ES: 進化戦略)、Differential Evolution (DE: 差分進化)、Firefly Algorithm (FA: ホタルアルゴリズム)を取り上げ、それらごとに有力な改良手法を網羅し、それらの統計的検定による性能評価を踏まえ、本論文の方法論の適用可能性を論じている。</p> <p>第3章においては、解きたい最適化問題にとって、メタヒューリスティクスのパラメータの最良調整則を求める問題を、メタ最適化問題として定式化し、この問題を GP の進化機能を用いてハイブリッド的に解くことで調整則を自動設計する方法論を説明している。</p> <p>つづく第4章においては、PSO, ES, DE, FA に対して、「メタ最適化」の考え方と手法を適用し、本論文で提案する方法論の有用性をやはり統計的検定によって検証している。とくに、これらの手法ごとに、最良の調整則を GP で獲得したのち、その関数を近似する簡単な表現式の導入によって新たな手法を提案し、それら手法間の壁を越える形で横断的かつ総合的な見地から、本論文で取り上げた手法の中で最高性能のものを選びすぐっており、第5章はその結辞である。</p>			

## SUMMARY OF Ph.D. DISSERTATION

School	Student Identification Number	SURNAME, First name Kanemasa, Minoru
<b>Title</b> Automatic Designing of Emergent Parameter Tuning Rule for Metaheuristics Using Genetic Programming		
<b>Abstract</b> <p>In recent years, many heuristic optimization algorithms capable of solving multi modal functions; metaheuristics, are proposed and used in many engineering problems. However, applying these methods come with problems of choosing appropriate method, and hyper parameters for a given algorithm and problem. This is because metaheuristics are not based on mathematical nor theoretical foundation. They are built based on experiences, simulations, and therefore, we don't have guarantees that the schemes, formulas, and preferred hyper parameters of an algorithm are optimal for a problem we want to solve. Besides, most metaheuristic algorithms are evaluated in an ideal environment, which we assume we can afford enough function calls. On the other hand, real world black box optimization problems tend to demand high computational power, thus we will not be able to calculate the objective functions enough to achieve their claimed performance in papers. Therefore, efficient development of good metaheuristic is desired.</p> <p>In this thesis, a computational way to automatically design a parameter tuning rule for metaheuristics is proposed, which actively controls the hyper parameter of an algorithm based on its current state. Insights from invariances of optimization algorithm are also incorporated, in order to discuss the effects of tuning rules and how they should be. Genetic Programming is used to solve the meta optimization problem formulated, and new emergent tuning rules are proposed for metaheuristics. The outline of this thesis is as follows.</p> <p>The first chapter will give an overview of the history of metaheuristics, and discuss the aim, purpose, structure of this thesis. In the second chapter, algorithms such as Particle Swarm Optimization (PSO,) Evolution Strategy (ES,) Differential Evolution (DE,) Firefly Algorithm (FA,) and their variants are broadly covered, and compared using statistical tests.</p> <p>The third chapter will focus on the core of this thesis. Designing parameter tuning rule for a metaheuristic is formulated as an optimization problem, and a way to solve this problem using Genetic Programming is shown.</p> <p>These methods are applied in the following 4<sup>th</sup> chapter. New tuning rules for the variants of algorithms proposed in the paper are obtained. Obtained rules are simplified and compared with other good algorithm designed by other researchers, in order to verify that computationally designed algorithms outperform human designed algorithms. Finally this research is concluded in the last chapter.</p>		