

論文の内容の要旨

論文題目 並列計算を利用した大規模スケジューリング問題の高速解法に関する研究

氏名 中村 友哉

本論文は、大規模なスケジューリング問題に並列計算を導入することによって、実用に耐えうる準最適解を高速に導出しようとするものである。

スケジューリング問題はNP困難問題の代表格として知られており、現在でもオペレーションズリサーチや計算量理論、人工知能といった分野において盛んに研究が行われている。スケジューリング問題を始めとするNP困難問題は問題のサイズとともに探索空間が組み合わせ爆発を起こすために、真の最適解を導出することは特殊な場合を除いて不可能である。そのため、近年では真の最適解の代わりに「最適にできるだけ近い解（準最適解）」を「実用的な時間範囲内で求める」ことが要請されるようになってきている。これを実現することは、実際に大規模スケジューリング問題を日々解かなければならない部品組立工場などの要望に沿うことにもなる。しかしこういった現場では次々と部品を投入し、制約違反が生じた場合にディスパッチングルールを適用することで解消するという非常にプリミティブな手法が主に用いられているのが現状である。

これを解決するためには、高速に準最適解を導出する手法が必要となる。従来、これを実現する手法としては遺伝的アルゴリズムや焼きなまし法といったメタヒューリスティクス的手法がよく研究されてきた。しかしメタヒューリスティクスは確率的な概念の応用や探索方法の工夫などによる広大な探索空間内の効率的に探索手法であり、得られる解の品質に対する理論的な背景がない。したがって、これらの手法では整数計画問題として定式化することのできるスケジューリング問題の特性を生かすことができない。

そこで本研究では上述のような探索的アプローチではなく、求解の際に最適性を陽に扱うことのできる理論的アプローチを用いる。理論的アプローチを実用的なスケジューリング問題に適用する際の課題は、計算時間である。準最適解導出手法であるから当然厳密解法より十分小さな計算時間でよいが、サイズの大きな問題においては実用的とはいえない時間が必要となることが多い。これを解決するため、本研究では並列計算の導入を提案する。具体的にはサイズの大きな問題を多数の子問題に分解し、それらの子問題を同時並列的に求解することによって高速化を図るというものである。この並列計算による計算時間短縮効果を期待できる準最適解導出手法として、ラグランジュ分解調整法を利用する。ラグランジュ分解調整法は得られた準最適解の品質評価性能を有する理論的アプローチの一手法であるが、問題を多数の子問題に分解し、それを独立に解くことで緩和問題の最適解を構成するという構造的な特徴（後述）を有し、並列計算を導入するにあたって非常に都合がよい。本手法を採用し、並列計算の導入による計算時間の短縮効果を見込むことで、相当大規模な問題でも十分実用的な時間内に計算を完了し、最適に近い解を出力できるようになると考えられる。

ラグランジュ分解調整法では、大規模なスケジューリング問題の一部の制約条件を緩和し、その上で多数の子問題に分割する。そして分解された子問題を独立に解き（最適化し）、それらを合わせたものを緩和問題の解とする。問題のサイズが小さい子問題を独立に解けばよいという意味で、緩和問題の解の導出の難しさを大幅に軽減できるところが画期的である。本来なら問題のサイズの増加に応じて指数関数的に計算量が増加するところを、線形増加にまで抑えることができるのである。

緩和問題の解は分解された子問題の解を合わせたものであるが、この解が実現する目的関数値はもとの問題の目的関数の下界値を与える（最小化問題の場合）。この下界値を与える解は各子問題が利己的に最適化した結果であるために通常は実行不可能であり、何らかの処理を行って実行可能スケジュールを生成する必要がある。これは、現在の実行不可能解の近傍を探索することで実現する。その解が実現する目的関数値はもとの問題の目的関数値の上界値とみなすことができる。最適解が実現する目的関数値は上界値と下界値の間にあるから、現在の解が最適解にどの程度近いかを定量的に評価することができる。これが最適性を陽に扱うことができるという主張の根拠である。

ラグランジュ分解調整法の最適解への漸近性能の調整パラメータとなっているのがラグランジュ乗数である。ラグランジュ乗数は制約条件を緩和する際に制約違反の度合いとの積として目的関数の中に取り込まれる。新しい目的関数（ラグランジュ関数）のもとでは、もとの問題の目的関数の最小化と制

約違反の度合いの最小化を同時に実現しようとするのであるが、このバランスを取るのがラグランジュ乗数である。この見方を変えると、もとの問題の最適解を求めることは最適なラグランジュ乗数を決定することであり、これは問題の双対性と関連がある。

ラグランジュ分解調整法は、子問題求解・下界値導出フェーズと実行可能化・上界値導出フェーズを繰り返すことにより最適なラグランジュ乗数を決定する（＝もとの問題の準最適解を与える）手法である。

大規模なスケジューリング問題をラグランジュ分解調整法によって解く場合には、分解される子問題の数が非常に大きいために下界値導出フェーズの計算時間がかかるのが通常である。そこで従来は、上界値導出フェーズにも探索に十分時間をかけ、精度の高い上界値を生成することでラグランジュ乗数の更新必要回数を減らし、イタレーション回数を抑えるという応用上の工夫がなされてきた。一方、本研究においては並列計算を導入することで処理の高速化を試みるのであるが、並列計算を適用できるのは実は子問題求解・下界値導出フェーズのみである。このため、並列計算のメリットを最大限に生かすためには下界値導出にかかる計算量の割合を十分大きく取る必要がある。そのために本研究では上界値導出フェーズはなるべく簡易的な方法で済ませ、下界値導出フェーズにより計算の軸足を移すと同時に、その下界値導出フェーズを並列計算の導入により超高速に処理することによって、計算時間の短縮を目指す。この方針ではイタレーション回数は増大するものの、最終的な準最適解を出力するまでの全体の計算時間は大きく圧縮される。

並列計算を導入するには、大きく分けて2通りの手法が存在する。1つがアルゴリズムレベルの並列化であり、もう1つが処理レベルの並列化である。アルゴリズムレベルの並列化（システム並列化）はスーパーコンピュータなど、汎用の並列計算機により実現することができる。一方処理レベルの並列化（ロジック並列化）は専用回路を設計した専用計算機により実現される。システム並列化可能な処理とロジック並列化可能な処理は互いに排他的ではなく、両方を満たす処理も両方とも満たさない処理も存在する。システム並列化を採用することによる高速化には限界があり、プロセッサ数 n に対し、 $1/n$ よりも処理を高速化することはできない。通常は全処理がシステム並列化可能ではないし、また複数のプロセッサに処理を分散することによるオーバーヘッドなどのために性能が落ちるからである。一方ロジック並列化では、高速化性能は回路の設計によるため、特に理論的な性能限界があるわけではない。ただし処理がハードウェア向きである必要があり、どのような処理でも簡単に回路化することができるわけではない。

本研究ではロジック並列化を利用して処理の高速化を図ることとし、重力多体問題の分野で成功を収めた専用計算機であるGRAPE-7の応用を想定する。GRAPE-7のプロセッサはFPGAであり、ユーザがカスタマイズしてロジックを設計することが可能であるために非常に低コストに導入が可能であり、また理想計算性能もスーパーコンピュータを凌ぐ。実験用のFPGAボードで得られた結果をもとに高速化性能を推算すると、問題によっては数千倍もの高速化性能を実現できることが示された。

以上の成果は、本論文で示したスケジューリング問題のみならず、特定の条件を満たす広い範囲の数理計画問題に対して適用でき、それらを必要とする現場への導入も比較的容易と思われることから、新しいスケジューリングツール、最適化ツールとしての普及が期待される。