

■日高宗一郎 アーキテクチャ科学研究系 助教

【タイトル】

プログラムの高速化を自動的に実現

【本文】

プログラムをより速く実行させる。これはコンピューターの性能を最大限に発揮させるだけでなく、コンピューターに囲まれて生活する現代において最も重要な技術課題の1つです。

プログラムとは、コンピューターに命令をするための“言語”で、それは私たち人間が使う言語と同じ構造をもっています。“言語”はたくさんの「要素」を組み合わせることで構成されます。そして、日本語に「話す」と「しゃべる」など同じような意味をもつ要素が複数あるように、プログラムに使われる“言語”にも同じ意味をもつものが複数あります。プログラムを速く実行させるためには、同じ意味をもついくつかの要素の中から最適なものを選び出し、組み込むことが重要になります。

通常、書かれたプログラムは、必ずしも最適な要素ばかりで構成されていないため、実行速度の改善余地が残っています。高速化にはプログラムの要素を最適なもの書き換えることが必要で、私はこれを自動的に行うことを目指しています。

プログラムの自動的な書き換え

例えば、「 X^2 」を計算させるプログラムがあります。このとき、「 X^2 」と厳密に同じ解答が得られるプログラムに「 $X \times X$ 」もあります。コンピューター上では「 $X \times X$ 」のほうが速く計算できるので、「 X^2 」を「 $X \times X$ 」に書き換えることでプログラムは速くなるわけです。

この研究でポイントとなるのは2つ。書き換える要素が本当に「同じ」かどうかと、書き換えるとどれくらい「速く」なるのかです。「同じ」かどうかは、要素ごとの変換規則を用意し、それが正しいかどうかを証明します。「速く」なるかどうかは、要素ごとに速度のモデルを用意し、それを組み合わせて書き換え前後のプログラムの速度差を見積ります。「 X^2 」の例を出しましたが、「 X^3 」と「 $X \times X \times X$ 」ならどうか、「 X^4 」ならどうかといったことを判断していくのです。このような準備をしておくことで、プログラムの自動的な書き換えが可能となります。

プログラムの速さを追求する

私はもともとプログラム変換に興味をもっていました。そして、大学院時代に出会った1つの論文が、私の研究に大きな影響を与えました。ある数学理論を駆使した論文で、要素の変換を自動的に行うというものでした。この考えをデータ処理向けの“言語”に当てはめ、今の研究があります。

実際に用いられるプログラムは非常に膨大・複雑で、要素をそれぞれ「速く」なるよう書き換えていけば、その効果は絶大です。しかし同時に、要素単位で速くなったからといってプログラム全体が本当に速くなっているのか、自動的な書き換えのための準備そのものが速さを妨げているかといった課題もみえてきます。

現在、プログラムの自動変換は、一部ですが可能となっています。今後さまざまな課題を克服し、実用的なプログラム自動変換システムを構築していこうと考えています。

(取材・構成 吉戸智明)