組合せ最適化

東京大学　松井知己　MATSUI Tomomi

1. はじめに

いままでに、組合せ最適化に興味を持った何人かの人々は、関連する本を紐解いてその途端に閉じてしまった経験があるだろう。それは、その本の中には数式がびっしり並び、かつまたその最初を読んでも、中に書いてあるのは「最短路問題」「最小木問題」「フロー問題」といった自分の抱えている問題とは似ても似つかない、あるいはあまりに単純過ぎる問題ばかりということが多いからである。この原因の一つは、これらの本がある種の問題を解くのに実用的に過ぎるからだと私は感じている。

そこでこの発表では、「組合せ最適化」を、問題を解くための便利な数学的ツールの集合ではなく、問題を解き方を考案するための枠組として捉えて簡単な紹介をしていきたいと思う。

2. 組合せ最適化問題

最適化問題にたどり着いた人が最初に抱えている問題の多くは「組合せ最適化」であることが多い。これは、これらの人の持つ問題が、（あることを）“やる”か“やらない”かといった離散的な性質を含んでいることが多く、また「一つ一つは単純な問題がいくつか組み合わさっている」といった風にある種の構造を内包ことが多いからである。この、離散的な性質を持ちある種の構造を内包しているということが組合せ最適化問題の大きな特徴である。

上のことは逆に、組合せ最適化問題は離散的な構造を許すため、問題の記述能力が高いという性質を表している。しかし問題が記述できることと、それが解けることはまったく別であることに注意しなければならない。また各問題は特有の構造を持つため、それを最大限利用することが望まれるが、全く同じ理由から、問題毎のアプローチが異なり一般的な議論が非常に困難となってしまう。

このように、組合せ最適化は記述能力の高さと解きにくさが表裏一体となっており、また個々のケースの成功が一般の議論に直接つながらない、という問題を本質的に抱えている。

3. 何故速く解けないといけないのか？

組合せ最適化の本や論文を開く時、まずぶつかるのがアルゴリズムの速さに関する記述である。いわゆるアルゴリズムの 「オーダー」である。一部には、組合せ最適化の研究はこのオーダーを小さくするのに血眼になっていると理解されている向きもある。しかし、そんなに速く解く必要が本当にあるのだろうか？

組合せ最適化で扱う問題では、その許容解はたかだか有限個しか無いのが普通である。例えば、東京から大阪に電車を乗り継いでできるだけ速く行くにはどうすればいいかという「最短路問題」を考えよう。この行き方は（同じ駅で 2 度おりるような無駄をしなければ）有限種類しかない。だからこれを全て探して一番速いものを探せば良い筈である。しかし、これは実際に不可能に近い。それは行き方の種類数があまりに多過ぎるからである。

すなわち、組合せ最適化問題を解くと言った時には、「実用的な時間で」という言葉が必ずついているわけである。その意味で、組合せ最適化においては、「速く解く」ことが本質的に切り離せない問題となっている。

これに対し、「将来コンピュータがより速くなれば、アルゴリズムの速さ（オーダー）の僅かな違いは関係なくなってしまうのではないか」といった議論をときおり耳にすることがある。しかしそれは完全な誤解であり、事実は全く逆である。

組合せ最適化において考慮されているアルゴリズムの速さとは、問題のサイズが 2 倍 3

倍になったとき、それを解くのにかかる時間は何倍になるのかということである。いま問題のサイズが 2 倍 3 倍になったら、それを解くのにかかる時間が 2 倍 3 倍になる（いわゆる $O(n)$）アルゴリズム A と 4 倍 9 倍になる（いわゆる $(O(n^2)$）アルゴリズム B があったとしよう。例えばコンピュータが今の１００倍速くなった時、アルゴリズム A では今と同じ時間で今の１００倍のサイズの問題を解くことが可能となるが、アルゴリズム B ではたった１０倍にしかならない。すなわち、この差はコンピュータの速度が速くなればなるほど大きな違いとして現れてくる。

4. 本当に最適なのか？

では今、ある組合せ最適化問題を　コンピュータで解いて解を得たとしよう。はたしてそれは、本当に最適解なのか？アルゴリズムが正しく、コンピュータプログラムが正確ならば当然答えはイエスである。ではその証拠は？

例えば、平面上にいくつかの点が与えられたとき、これを全て内部に含むような半径最小の円を求める問題を考えよう。この問題の最適解の円の中心は、その周上の 3 点が作る三角形に含まれることが分かる。このときは、この三角形が証拠となる。

現在効率良く解ける組合せ最適化問題のほとんどは、このような証拠が存在することが知られている。例えば、線形計画問題においてはその双対問題の最適解がその証拠と考えることができる（線形計画問題は連続な変数を扱う問題ではあるが、組合せ最適化問題の代表的な例の１つでもある）。

この“証拠”の数学的な定義は非常に難解なものであるが、現在この証拠があることが、その問題に効率的な解法が存在することの手がかりとなっていると考えられている（数学的な言葉を用いるならば、NP かつ co-NP の問題には多項式時間解法があると予想されている）。また多くの場合、この証拠は求める最適解が満たすべき条件として現実的な解釈を与えることができる。発表ではいくつかの問題の例を引いて、その証拠が何になるのか、またそれの現実的な解釈はなんなのかを考えてみたい。

5. それしか手はないのか？

さて、最適解が得られたとして、それを使おうとしたとき起こるのが「それしか無いのか？」という疑問である。最適解が複数あるのならば、その中でも更にいいものを求めるのが普通であろう。また、そもそも最適化問題では単一の目的関数しか扱わないが、実際は「これが満たされたら次はあれ」と次々と要求が出てくることが多い。

組合せ最適化問題においては許容解の個数が有限個であることから、これを順に並べあげるというプリミティブなアプローチが最近研究されている。その一つは最適解を全て求めるというもの。もう一つは、解を良い方から順にあげるというものである。特に後者は（良いものから順に K 番目まで求めるという意味で）K-best 問題と呼ばれている。

組合せ最適化によって現実問題が完全に解けるということは、一般に非常に少ない。大量のメモリと高速な演算が可能となりつつある現在においては、上記のようなアプローチによって得られた複数個の解から意思決定者が望ましい解を取捨選択していくという形の利用方法が増して行くのではないかと予想される。

6. おわりに

「難しい」「使えない」といった声を良く聞くことがある　組合せ最適化であるが、この原因の一つには、使いやすいソフトウェアパッケージが無いという理由が考えられる（例えば現在、最小費用流や巡回セールスマン問題を高速に解くソフトウェアが必要なとき、各言語についていったい何種類あるだろうか？）。このようなアルゴリズムやソフトウェアの開発に対し、研究者や企業の協力による組織的な研究体制が現在必要となっていると考えられる。更に加えるならば、個々の効率的に解ける問題（最小費用流問題や巡回セールスマン等）は逆にブラックボックスとしてそれをどう組み合わせるかという問題を真剣に（できれば理論的に）検討すべき時が来ているのではないだろうか。