

Web サーバを利用したメタ戦略の並列実行システムの試作

川端 宏明 (沼田 一道 助教授)

1. はじめに

組合せ最適化問題 (与えられた制約条件下で目的関数値が最良になる組合せを求める問題) は実行可能解の数が有限個であり, 全解列挙することで厳密な最適解を求めることができる. しかし, 問題の規模が大きくなるにつれて列挙数が爆発的に増加するため, 実用的な計算時間で厳密な最適解を求めるのは困難である. そのために近似解法である様々な発見的解法 (Heuristics) が提案されてきた. メタ戦略 (Meta Heuristics) とは, この発見的解法を用いた解の精度を向上させるための一般的な枠組みの総称である. しかし, メタ戦略は手軽に探索できる反面, 発見的解法と比べるとかなりの実行時間を要する. そこで本研究では, 複数台のコンピュータを協調させ, メタ戦略を並列実行することによって実行時間の短縮を図る.

本研究ではその際に Web サーバを利用することを検討する. Web サーバとは Web 情報を配信することに特化したコンピュータのことであるが, クライアントの要求に応じて動的な処理を行う機構も持っている. この機構を利用してメタ戦略を並列実行し実行時間の短縮を図ること, そして Web サーバがこのような並列実行システムに有用かどうかを評価することが本研究の目的である.

2. 本研究で取り扱う問題と解法

本研究ではメタ戦略として, 頑健で高精度であるがやや時間のかかる RSB 法 (Reduction by Selection Bias) を採用する. RSB 法は独立な局所最適解を多数生成させる手法なので並列計算に適していると考えられる. また対象とする問題としては, 標準的な組合せ最適化問題とされる巡回セールスマン問題を採用し, 発見的解法には 2-opt 法, 3-opt 法を採用することにする.

2. 1 巡回セールスマン問題

巡回セールスマン問題 (Traveling Salesman Problem : TSP) は, 与えられた都市をすべて訪問するとき, 距離や時間に基づく総移動費用が最小となる巡回路を求める問題である. 巡回セールスマン問題は以下のように定式化される.

- n : 都市数 S_n : $\{1, \dots, n\}$ の順列全体の集合
 $\sigma(i)$: 順列 σ において, i 番目に訪問する都市 (i, \dots, n) ここで, $\sigma(n+1) = \sigma(1)$ とする
 C_{ij} : 都市 i, j 間の移動費用 ($C_{ij} = C_{ji}$, $i, j = 1, \dots, n$, $i \neq j$)
 Z : 巡回路の総移動費用

$$\text{Minimize } Z = \sum_{i=1}^n C_{\sigma(i)\sigma(i+1)}, \quad \text{Subject to } \sigma \in S_n$$

2. 2 RSB 法

RSB 法 (Reduction by Selection Bias) とは, 多数発生させた局所最適解の上位何個かを選び, その共通部分を「最適解の構成要素」とみなして固定し, 問題のサイズを縮小することを繰り返す戦略である. RSB 法の動作は以下のように書ける.

- Step1: 問題のサイズを $s := n$ とする (n : 都市数)
- Step2: ランダムに選んだ初期解を逐次改善して局所最適解を求めることを L 回行う
- Step3: L 個の局所最適解から目的関数値の良いものを k 個選び出す
共通部分がなければ, Step2 に戻る
- Step4: しかるべき長さ以上の共通部分を最適解の構成要素とみなして固定し,
問題のサイズを縮小する (s を更新する)
- Step5: s が十分小さくなったら Step6 に進む, そうでなければ, 縮小された問題を
新たな問題として Step2 に戻る
- Step6: 元の n 都市の問題に戻し, 今までに得られた最良の解を出力する

2. 3 2-opt 法, 3-opt 法

2-opt 法とは, ある巡回路においてある 2 本の枝を取り除き, 別の 2 本の枝を付け加えて再び巡回路を作ることを考える. このとき, 巡回路の長さが短くなる場合にはこの入れ替えを行い, 改善されない場合には行わない. この操作を巡回路の長さを減少させる入れ替えがなくなるまで繰り返す方法である. (図 1)

同様に, 3 本の枝の入れ替えを考えるのが 3-opt 法である. (図 2)

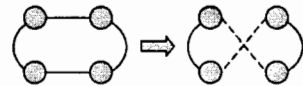


図 1 2-opt 法

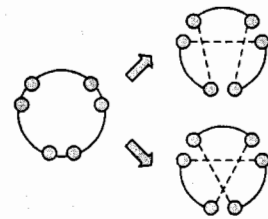


図 2 3-opt 法

3. システムの概要

3. 1 本システムについて

本システムはネットワークで結ばれた複数台のクライアントマシンと 1 台のサーバマシン (図 3) で構成され, 複数のクライアントプログラムを協調させるために Web サーバを利用している. Web サーバはサーバソフトウェアが常駐しているのでいつでもクライアントの要求に応えることができるほか, 必要に応じてサーバプログラム (CGI プログラム) を起動し, 動的な処理を行うこともできる. このような既存のしくみを利用することで複数のプログラムの制御を容易にできることが Web サーバを用いる最大のメリットである. また, Web サーバはこのように完全に受身であるので, いつでもクライアント主導でシステムを実行できる利便性もある. 以下にマシン環境と各プログラムの輪郭を示す.

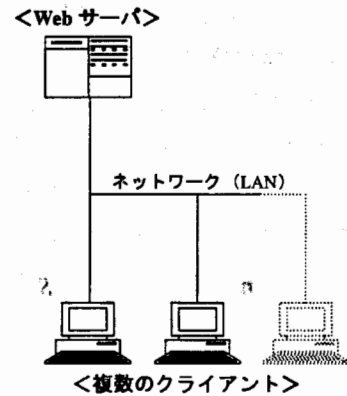


図 3 動作環境図

クライアント	Webサーバ
<input type="checkbox"/> マシン : PC/AT互換機 <input type="checkbox"/> OS : Windows98/Me/2000/XP (ブラウザが動作するマシンであればよい) <input type="checkbox"/> クライアントプログラム : Delphi5 Professional (言語)	<input type="checkbox"/> マシン : PC/AT互換機 <input type="checkbox"/> OS : Windows2000 <input type="checkbox"/> サーバソフトウェア : IIS (Internet Information Server) <input type="checkbox"/> サーバプログラム : Delphi5 Professional (言語)

クライアントプログラム : システムの開始や終了, 局所最適解の生成, 送信等を行う
(Delphi5のインターネットコンポーネントを利用)

サーバプログラム : クライアントからの結果を元に問題の縮約を行う

3. 2 動作概要

本システムでは複数のクライアントマシンの中から任意に1台を選択し、システム実行の開始や終了、待機中の他のクライアントの監視や操作、情報統制を行うことが可能である。このとき選択したクライアントマシンを「クライアントヘッド」と呼ぶことにする。以下に、本システムの動作概要を示す。

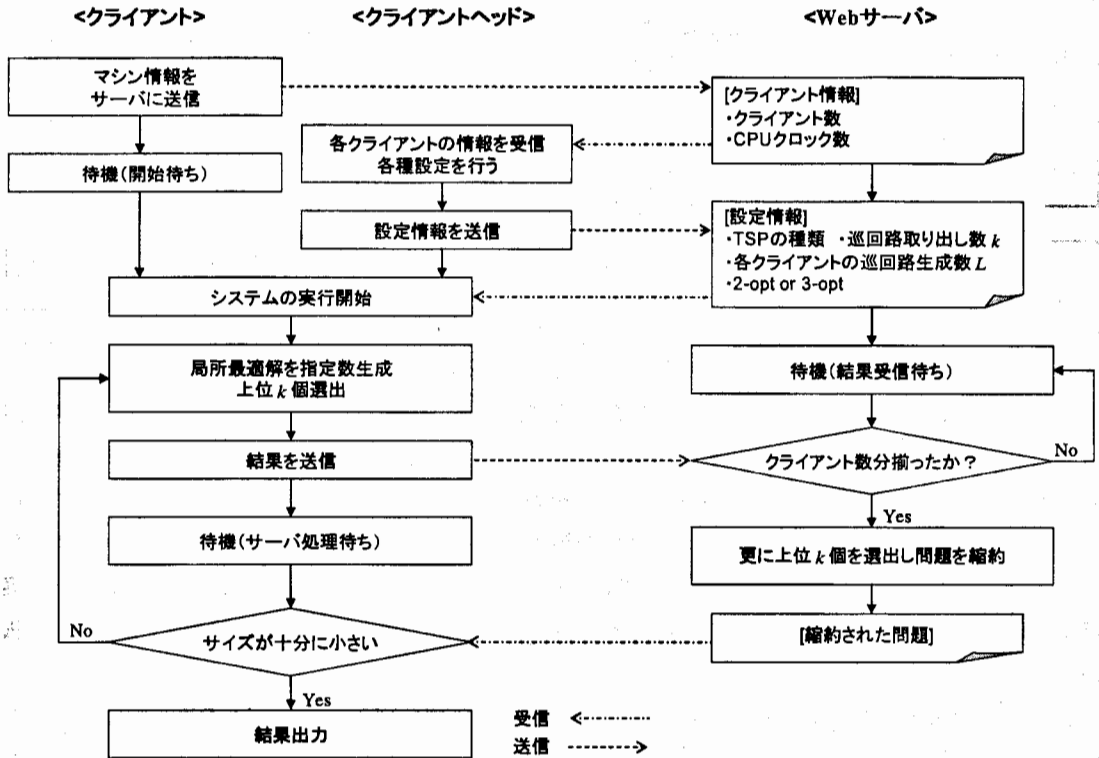


図4 各クライアントと Web サーバの状態遷移

4. 数値実験

4. 1 実験方法

TSP のベンチマーク問題集である TSPLIB95 を使用して数値実験を行い、並列実行による実行時間の短縮の確認や、発見的解法 (2-opt 法, 3-opt 法) の違いによる実行時間と精度の比較を行う。

また図4より、本システムは Web サーバが各クライアントからの結果がすべて揃わないと、縮約作業に移ることができないので、巡回路生成数を均等に配分した場合、一番低速のマシンに実行速度を合わせることになる。このことを考慮し、マシンによって CPU 速度が異なる場合、各マシンの処理速度を測る予備実験を行うことでより良い配分を割り出し、等配分の場合の実行時間と比較する実験も行う。

4. 2 実験結果

実験結果を以下の表に示す。準最適値、実行時間は 10 回の試行の平均による。クライアント数欄のカッコにはそのマシンの CPU クロック数を、準最適解欄のカッコには最適解が得られた回数をそれぞれ示した。

表 1 実験結果 (1)

問題名 (最適値)	クライアント数 ([MHz])	解法 (生成数 L , 取出数 k)	準最適値	実行時間[s]	短縮時間比率
lin105 (14379)	1 (1000)	2-opt (500,7)	14381 (9)	15.529	—
lin105 (14379)	2 (1000,1000)	2-opt (250/250,7)	14382 (9)	9.865	0.64
lin105 (14379)	3 (1000,1000,1000)	2-opt (167/166/166,7)	14379 (10)	7.458	0.48

表 2 実験結果 (2)

問題名 (最適値)	クライアント数 ([MHz])	解法 (生成数 L , 取出数 k)	準最適値	実行時間[s]	短縮時間比率
eil51 (426)	1 (1000)	2-opt (150,3)	427.4 (2)	0.393	—
eil51 (426)	3 (1000,1000,1000)	2-opt (50/50/50,3)	427.0 (2)	2.601	6.62
eil51 (426)	1 (1000)	3-opt (150,3)	426 (10)	12.052	—
eil51 (426)	3 (1000,1000,1000)	3-opt (50/50/50,3)	426 (10)	5.385	0.45

表 3 実験結果 (3)

問題名 (最適値)	クライアント数 ([MHz])	解法 (生成数 L , 取出数 k)	準最適値	実行時間[s]	短縮時間比率
lin105 (14379)	3 (1000,450,300)	2-opt (100/100/100,7)	14379 (10)	13.77	—
lin105 (14379)	3 (1000,450,300)	2-opt (175/75/50,7)	14379 (10)	8.373	0.61

4. 3 考察

表 1 は同一マシン 3 台を用いて行った実験結果である。この表から、並列実行による時間短縮が確認できる。しかし、単純に短縮時間比率が $(1/\text{クライアント台数})$ にはなっていない。これは、Web サーバが中間生成物 (テキストファイル) を多く作るために生じるタイムロスの影響である。

表 2 は各解法の精度と実行時間を比べた実験結果である。この表では 3-opt 法が実行時間はかなりかかるものの、10 回とも最適値を示しており精度の高さがうかがえる。また、3-opt 法は並列実行することにより実行時間がかなり短縮された。しかし、実行時間がほとんどかからない 2-opt 法の場合は、中間生成物によるタイムロスの影響が大きく並列実行のほうが悪い結果になった。

表 3 は巡回路生成数の配分を替えることで実行時間の短縮を図った実験結果である。この表より等配分するよりも、予備実験を行いその結果を参照して配分したほうが、実行時間を短縮できることがわかる。この方法は、Web サーバの処理を待つタイムロスを軽減させるために有効である。

5. まとめ

本研究では、複数台のコンピュータを並列実行させることで、RSB 法の実行時間の短縮を試みた。この方法により、かなりの時間が短縮され、並列実行の有効性が確認できた。

また、本研究では複数のクライアントを容易に協調させるために Web サーバの利用を試みたが、クライアントとのやり取りをする際に、中間生成物を多く作るために生じるタイムロスが、場合によっては無視できないものであることもわかった。しかし、問題の規模が大きいときや 3-opt 法のように時間のかかる解法を用いるときには、このタイムロスの比重は小さくなるので、Web サーバを用いた柔軟性のある並列実行システムは有望であると思われる。

【参考文献】

- [1] 沼田 一道, 児玉 淳: 「解の淘汰により変数の固定を行う解探索について」, システム制御情報学会論文誌, vol.12, p57-59, 1999.
- [2] 西村 めぐみ: 「Delphi5 で作る Web アプリケーション入門」, 広文社, 2000.