

シミュレーテッドアニーリング概論

An outline of Simulated Annealing

SA 研究グループ

SA Team

Abstract: This paper introduces the mechanism of Simulated Annealing(SA) and Temperature Parallel Simulated Annealing(TPSA). Simulated Annealing is an algorithm to solve optimization problems. It has the advance of escaping from local optima. However, it requires long computation time, and it is extremely difficult to determine proper cooling schedules which control behaviors of solutions.

1 はじめに

SA¹⁾ の基礎となる考えは Metropolis らが 1953 年に発表した焼きなましと呼ばれる過熱炉内の固体の冷却過程をシミュレートするアルゴリズムに端を発し、最適化問題、特に組み合わせ最適化問題を解く汎用近似解法の 1 つとして用いられている。SA は、局所探索をランダムに行いながら、更に解に改良が見られない場合でも、新しい解に移る可能性を残すことで局所解に陥ることを防ぐことができる点に特徴がある。この特徴により、現在では組合せ最適化問題にだけでなく、複数の局所解を持つ連続変数最適化問題にも用いられている。しかし、SA には解を得るまでの時間が長いという欠点があり、巡回セールスマン問題では SA で良好な近似解を得る計算量よりも完全な総当り計算の方が計算量がすくないことが報告されている²⁾。そのため、近年 SA の計算負荷を軽減するために、並列化の研究が盛んに行われてきた。しかしながら、いずれの方法も SA のクーリングスケジュールが経験的にしか与えられていないという問題は常に残る。

これに対して、温度並列 SA³⁾ は並列処理との高い親和性を持っているだけでなく、温度スケジュールが不要であるという極めて優れた特徴を有している。

本発表では、SA の基本アルゴリズムと TPSA のアルゴリズムについて説明する。

2 アルゴリズム

SA のアルゴリズムの流れを Fig. 1 に示す。SA のアルゴリズムで重要になるのが、生成処理、受理判定、クーリング処理である。

2.1 生成処理

生成処理では、現在の状態 x を与えられて次に推移すべき状態 x' を生成する処理である。この生成処理には、

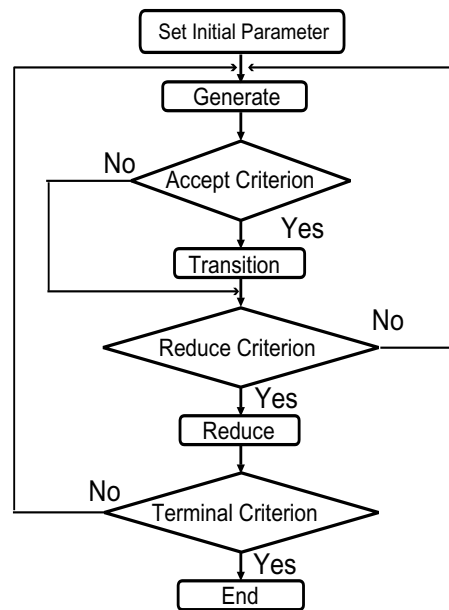


Fig. 1 SA のアルゴリズム

状態 x が与えられて状態 x' が生起する確率分布 $G(x, x')$ を用いる。組合せ最適化問題の場合は、状態 x' は状態 x の近傍にあたり、推移に優先性を与えない場合は式 (1) のような等確率推移となる。ここで、 $n(x)$ は、状態 x の近傍を構成する状態の数を表す。

$$G(x, x') = \frac{1}{n(x)} \quad (1)$$

2.2 受理判定

受理判定は、次の状態 x' のエネルギー E' と現在の状態 x のエネルギー E との差分 $\Delta E (= E' - E)$ 、および温度パラメータ T によって、次の状態への推移を受理するか否かの判定をする。通常は式 (2) の Metropolis の基準^{?)} が採用される。

$$A(E, E', T) = \begin{cases} 1 & \text{if } \Delta E < 0 \\ \exp\left(-\frac{\Delta E}{T}\right) & \text{otherwise} \end{cases} \quad (2)$$

2.3 クーリング処理

SA は、生成処理と受理判定を繰り返すアルゴリズムである。そしてある程度の繰り返しを行った後にクーリングを行う。クーリングは第 k ステップの温度 T_k を与えて、次のステップの温度 T_{k+1} を返す処理のことである。最適解への漸近収束性を保証するには急速に冷やしてはならないが、実時間で計算を行うために真の最適解への収束を犠牲にして、式 (3) に示す指数型アニーリングがよく使われる。

$$T_{k+1} = \gamma T_k \quad (0.8 \leq \gamma < 1) \quad (3)$$

2.4 終了条件

終了条件には以下のような実装方法がある^{?)}。

- ・アニーリングを定めた回数だけ繰り返して終了
- ・受理がほとんど起こらなくなると終了
- ・同じ状態が何度も生成されるようになると終了
- ・温度が十分低くなると終了
- ・エネルギーの変化、またはエネルギー自体が十分小さくなると終了

3 SA の長所と短所

● 頑強性

多くの最適化解法が局所最適解に捕捉される欠点を持つのに対し、SA は容易に捕捉されず、理論上は真の最適解に、実際には準最適解に到達できる。これは、解品質が改良方向のみ探索を進めるのではなく、時折、改悪する方向も選ぶ仕組みによる。

● 汎用性

枠組み自体が極めて汎用にできているので、実に広範囲の問題に適用できる。

● 柔軟性

目的関数 (コスト関数) に対する制約がほとんどない。滑らかさ、連続性、決定性はいずれも満たされなくてもよい。つまり、目的関数は微分可能でなくても、複雑な式で求まるものであっても、確率的であってもよい。さらに、問題に複雑な境界条件があってもよい。

● 簡便性

アルゴリズムはきわめて簡単で誰でも容易に作れる。

また、SA には以下の欠点もある。

● 非効率性

最適解を得るのに非常に多くの計算量を要する。この問題を克服するため、逐次処理のまま高速なアニーリングを導入する高速化の研究、および並列化して高速化を図る並列化の研究が近年見られる。

● 操作性

汎用解法であるため、特定の問題を解く場合には、パラメータをチューニングする必要がある。特に、温度と呼ばれる制御パラメータのチューニングが極めて困難となる。

4 TPSA の基本アルゴリズム

逐次 SA では式 (3) で示す割合で温度を冷却するのに対して、TPSA のアルゴリズムでは温度 T のプロセスから温度 T' のプロセスへ解を渡す。この時解交換は確率的に行う。逐次 SA で温度スケジュールを設定することは、TPSA ではプロセッサ間の解交換に相当する。つまり TPSA では、プロセッサ間で解交換を確率的に行わせることによって温度スケジュールを自動化することができる。すなわち、確率的な解交換によって適した温度スケジュールを選び出すことができる。

5 TPSA の利点

● 温度スケジュールの自動化

温度スケジュールを解が自分自身で決定する。

● 時間一様性

ある時点で処理を打ち切って得られた解が不満足なものならば、処理を継続してさらに最適化を図ることができる。

● 並列処理との高い親和性

並列処理で効率が抑制される原因の一つにプロセッサ間通信が考えられる。TPSA では、プロセッサ間通信が必要となるのは解交換の瞬間のみであるために並列処理に適したアルゴリズムである。

参考文献

- 1) Kirkpatrick, S., Gelatt Jr. C.D., Vecchi, M.P.: Optimization by Simulated Annealing. Science, 1983.
- 2) Aarts, E., Korst, J.: Simulated Annealing and Boltzmann Machines. John Wiley & Sons, 1989.
- 3) 小西 健三, 瀧 和男, 木村 宏一. 温度並列シミュレートドアニーリング法とその評価. 情報処理学会論文誌, 1995.