

シミュレーテッドアニーリング概説

An outline of Simulated Annealing

池内 智悟, 窪田 耕明

Motonori IKEUCHI, Komei KUBOTA

Abstract: Simulated Annealing is an algorithm to solve optimization problems. It has the advantage 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. In this paper, two approaches are considered for extended algorithms of SA.

1 はじめに

SA¹⁾ 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個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, 特に組合せ最適化問題を解く汎用近似解法の一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のための, 多くの SA アルゴリズムも提案されている. しかし, SA は解を得るまでの計算時間が長いという欠点を持っている. これは SA の最大の欠点でもあり, 例えば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ほうが計算量が少ないことが報告されている.²⁾ また, SA は組合せ最適化問題においても連続最適化問題においても, 温度と呼ばれる制御パラメータを持っていて, この温度を下げすぎると局所解から脱出できる確率が低くなるので, 解探索の振る舞いを制御する冷却スケジュールの決定が非常に困難という欠点も持っている. そのため, 近年, SA の計算負荷を軽減すべく, 並列化の研究が盛んに行われてきた. それらの中には温度スケジュールが原理的に不要であるという温度並列シミュレーテッドアニーリング (TPSA) というものもある.³⁾

本研究では, PC クラスタと呼ばれる並列計算機への並列 SA の実装方法や TPSA の温度設定を詳しく検証することで, SA の改良のための指針を得ることを目的とする.

2 シミュレーテッドアニーリング

物質を融解状態になるまで加熱し, 徐々に冷却する操作を焼きなまし (アニーリング) という. この焼きなましにより, エネルギーが最も少ない状態に分子が配列し, 結晶構造を形成させることができる. この物理プロセス

に着想を得て, これを計算機上で模擬することにより最適化問題を解こうとする手法をシミュレーテッドアニーリング (SA) と呼んでいる.

2.1 基本アルゴリズム

SA の基本アルゴリズムを Fig. 1 に示す. まず温度 T を初期設定した後, 与えられた状態 x_0 から出発して次の状態 x' を生成し, そのエネルギー E' を計算する. エネルギーの差分 $\Delta E (= E' - E)$ と温度 T_k に応じて受理するか否かを計算し, 受理の場合は次の状態に推移する. この処理を繰り返して, 現在の温度での平衡状態が実現されるまで十分な探索を行う. 温度 T_k で平衡状態に達したら, 徐冷処理を行って次の温度 T_{k+1} を求め, 再びその温度で平衡状態に達するまで十分な探索を進める. 十分温度が冷えて終了条件に達すれば, そのときの状態とエネルギーをおおのこの最適状態, 最適値として出力する.

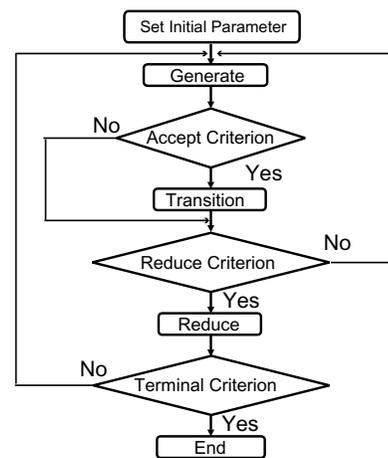


Fig. 1 SA のアルゴリズム

2.2 SA の長所, 欠点

SA には以下のような長所がある.

- SA は容易に局所解に捕捉されず, 理論上は真の最

- ほとんど任意の制約条件，境界条件を処理できる。
- 目的関数に対する制約がなく，微分可能でなくても複雑な式であっても，確率的であっても良い。
- コード化がきわめて容易である。

また，SA には以下の欠点もある。

- 最終的に最適解に到達することが保証されているが，そのためには膨大な量の計算が必要とされる。
- SA において最適化性能を左右する温度パラメータの設定が極めて困難である。

3 シミュレーテッドアニーリングの並列化

シミュレーテッドアニーリング (SA) には，良質な解を得ようとすればするほど，膨大な量の計算が必要になってくるといふ問題がある。こういった非効率性を克服するための方法は二つ挙げられる。一つは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⁴⁾，つまり，より早く処理できるアルゴリズムの研究である。もう一つは並列化して高速化を図る並列処理の研究²⁾である。以下では後者の研究について述べる。

逐次 SA の非効率性を解決するためのアプローチとして，Aarts と Korst は以下のような並列 SA を明らかにしている。²⁾

- 完全独立型
- 適応型
- 定期同期型
- 非同期型
- 受理時同期型

その他にも，Systolic 型⁵⁾と呼ばれるものがある。

上に述べられているものは逐次 SA の欠点の一つである，膨大な計算量を克服するための手法として考えられた並列 SA 手法である。

4 温度並列シミュレーテッドアニーリング

4.1 TPSA の基本アルゴリズム

逐次 SA では温度 T を温度 T' に減少させるのに対して，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。このときの解交換は確率的に行う。逐次 SA のアルゴリズムで温度スケジュールを設定することは，TPSA のアルゴリズムではプロセッサ間の解の交換をいつ行うかを指定することに相当する。TPSA では，プロセッサ間の解の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している。すなわち，確率的な解の交換によって，解自身が自分に適した温度スケジュールを選び出してくれることを期待するのである。

TPSA は逐次 SA と比較して，以下の利点がある。

- 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。
- アルゴリズムの時間一様性

TPSA は，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大きな原因の一つに，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。

5 昨年度の研究成果

5.1 PC クラスタにおける PSA の検討

これらの中より PC クラスタに実装するための並列 SA を選択した。以下に挙げるものがその手法である。

完全独立型：最も単純で，プロセッサごとに逐次 SA を互いに独立に実行させる方式である。推移は正当で漸近収束も保証される。

定期同期型：複数のプロセッサで一定期間互いに独立に SA を実行し，その後同期をとって全体の中で最も良い解を求め，再び全プロセッサがその解から次の期間の処理を再開する方法である。これは同期をとるため推移は正当で，漸近収束も保証される。

受理時同期型：複数のプロセッサに SA を並列に実行させ，いずれかのプロセッサで受理が起こったならば，全プロセッサの途中処理を破棄し，受理した推移を実行し，全プロセッサが次の状態から並列処理を再開する方式である。同期をとるので推移は正当で漸近収束も保証される。

Systolic 型：長さ L のマルコフ連鎖をプロセッサ数の P 個に分割し，プロセッサは長さ L/P のマルコフ連鎖を担当する。 L/P の SA を実行した後，プロセッサ内でクーリングを行う，または異なったプロセッサで同じ温度を持つマルコフ連鎖を実行する。異なったプロセッサから解を受け取るプロセッサは，自分の解で SA を実行するか，受け取った解で SA を実行するかどうかを確率的に決定する。

上に挙げた並列 SA の中で，PC クラスタにおいて定期同期型および受理時同期型を連続設計変数空間最適化問題 (rastrigin 関数) へ適用したところ最適解は得られなかった。

5.2 TPSAの温度設定に関する考察

TPSAは並列処理との親和性が高く、温度スケジュールを自動化できるという特徴を持ち、これまで多くの問題に応用され、逐次SAよりも計算時間および解の精度の点で優れていることがわかっている。しかし、それらの研究においては、温度パラメータの設定は固定的なものであり、その影響については述べられていない。

本研究では、TPSAを巡回セールスマン問題(TSP)に適用し、温度パラメータが解に与える影響を検証し、TPSAにおける温度に関する問題点を明確にし、TPSAの改良のための指針を得ることを目的とした。

5.2.1 重要な温度

TPSAの各温度の中で、一番良い探索をする温度(以後、重要な温度と呼ぶ)が存在していると考えた。この重要な温度が存在するかどうかを調べるため、最高温度、最低温度を従来のTPSAと同じ方法で決定し、32個のプロセスに等比的に温度を振り分け、各プロセスは与えられた温度で独立に探索を行い、各プロセスが求めた解の精度を比較した。結果をFig. 2に示す。横軸はプロセス番号であり、プロセス1が最低温度を担当しているプロセスである。なお、プロセス20より高温のプロセスは省略してある。

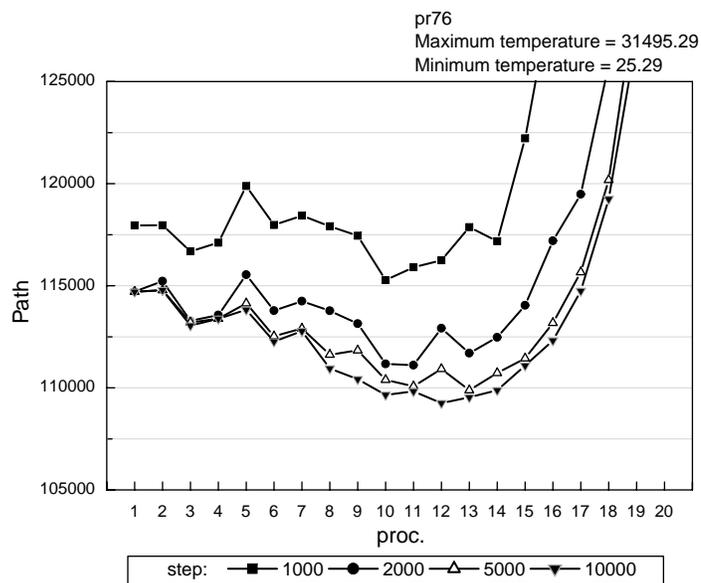


Fig. 2 各プロセスが独立に処理を行ったときの解の精度

Fig. 2より、重要な温度はプロセス10(200度を担当するプロセス)からプロセス14(500度を担当するプロセス)の間に該当することがわかる。

5.3 3つのアルゴリズムの比較

ここで、以下の3つのアルゴリズムを用意し、比較検討を行った。

1. 従来のTPSA

2. 温度調節を行ったTPSA

3. 重要温度に固定したPSA

アルゴリズム2は、最高温度と最低温度の差を狭めたものでアルゴリズム1よりも重要な温度を持つプロセス数が多くなる。アルゴリズム3は、Fig. 2で最も良い精度の解を出力したプロセスの温度を32個のプロセス全てに与え、解交換を行わずに処理させ、最後に一番精度のよい解を出力する。これはクーリングを行わず、32個のプロセスが同温で独立に2-opt交換とMetropolis基準による受理判定を行っていくアルゴリズムである。

この3つを比較するとアルゴリズム3が最も良い性能を示した。この結果により、SAでは重要温度の状態での探索のみが解を得るために必要な処理であるということ、また、重要温度を担当しているプロセス数が多い方が性能が良いということがわかった。

6 今後の課題

6.1 PCクラスタにおけるPSAの検討

- PCクラスタにおける並列SAの組合せ最適化問題(巡回セールスマン問題)への適用
- よりPCクラスタに有効である手法の提案

6.2 TPSAの温度設定に関する考察

- 重要な温度のより厳密な調査
- 解の精度が良質な方向に温度を調節して重要な温度を見つけ、その時点で決定した重要な温度とそのプロセスの解を全てのプロセスに与えて、続きの処理を行う適応的なTPSAの提案

参考文献

- 1) Kirkpatrick, S., Gelett 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テッドアニーリング法とその評価. 情報処理学会論文誌, 1995.
- 4) Szu, H., Hartley, R. *Fast Simulated Annealing*. Physics Letters A, 1987.
- 5) D.R.Greening. *Parallel simulated annealing techniques*. Physica, 1990.