

# カオスに基づく大域的最適化に関する研究

80122266 須崎 公介 指導教員 志水 清孝

## 1. 概要

関数の大域的最適化アルゴリズムは現実世界の科学的、工学的な問題から生じる要求を満足させることができるため、その研究成果は大変必要とされている。

本論文では、まず、直接勾配降下制御と呼ばれる制御理論の応用から得られる動的な非線形計画法について説明し、そして次にその分析をおこなう。そしてその分析をベースに、新しい無制約多峰性関数の大域的最適化手法を提案する。

## 2. 動的な非線形計画法

動的な非線形計画法 (DNP) とは評価関数の時間変化率  $\frac{dF(x)}{dt}$  をの情報を動的なコントローラに組み込む最適化手法で、動的なコントローラ  $u(t)$  を制御入力とみなすことで、静的な最適化問題を動的な制御問題として定式化する。

評価関数  $F(x)$  を最小にする最適化問題を解くにあたって、関数値  $F(x)$  を局所的に減少させるような減少化問題

$$\text{decrease } F(x(t)) \quad (1)$$

を考える。一方、一般の減少化制御問題は

$$\text{decrease } F(x(t), u(t)) \quad (2)$$

$$\text{subj.to } \dot{x}(t) = f(x(t), u(t)), x(0) = x_0$$

と定式化される。(2) の解法として制御ベクトル  $u$  の空間における最急降下法である、直接勾配降下制御 (DGDC) の理論を用いると、 $u(t)$  の修正式は次式で与えられる。

$$\dot{x}(t) = u(t), x(0) = x_0 \quad (3)$$

$$\dot{u}(t) = -(f_u(x(t), u(t))^T F_x(x(t), u(t))^T + F_u(x(t), u(t))^T), u(0) = u_0 \quad (4)$$

ここで  $\frac{dF(x(t))}{dt} + F(x(t))$  を減少化する次のような減少化制御問題を考える。

$$\text{decrease } \frac{dF(x(t))}{dt} + F(x(t)) + \frac{1}{2}\gamma u(t)^T u(t) \quad (5)$$

$$\text{subject to } \dot{x}(t) = u(t), x(0) = x_0 \quad (6)$$

記述簡単化のため次のようにおく

$$\begin{aligned} \phi^t[u(t)] &\triangleq \hat{F}(x(t), u(t)) \\ &\triangleq \frac{dF(x(t))}{dt} + F(x(t)) + \frac{1}{2}\gamma u(t)^T u(t) \\ &= F_x(x(t))u(t) + F(x(t)) + \frac{1}{2}\gamma u(t)^T u(t) \end{aligned}$$

$$f(x(t), u(t)) \triangleq u(t)$$

ここで式 (4) のように勾配関数  $\nabla \phi^t[u](t)$  を計算すると  $f_u(x(t), u(t)) = I$  より

$$\begin{aligned} \nabla \phi^t[u](t) &= I \cdot \hat{F}_x(x(t), u(t))^T + \hat{F}_u(x(t), u(t))^T \\ &= 2F_x(x(t))^T + F_{xx}(x(t))u(t) + \gamma u(t) \quad (7) \end{aligned}$$

よってアルゴリズム 1 は次のように定義される。

$$\dot{x}(t) = u(t), x(0) = x_0 \quad (8)$$

$$\begin{aligned} \dot{u}(t) &= -\alpha(2F_x(x(t))^T + F_{xx}(x(t))u(t) \\ &\quad + \gamma u(t)), u(0) = u_0 \quad (9) \end{aligned}$$

以上あげたアルゴリズム 1 は単峰性関数の最適化問題に対する最適性だけでなく、多峰性関数においても一部の軌道は局所解の谷を抜け大域的最適解へと向かうことが確認されている。

## 3. 動的な非線形計画法の分析

動的な数値計画法はシミュレーションの結果、局所最適解の探索の優れた能力を有するのみならず、時にはある程度の大域的最適解の探索を行う機能を有することことが観察された。そこで、このアルゴリズムの分析をこれよりおこなう。

まずアルゴリズム 1 の特性を調べよう。アルゴリズム 1 の (9) の右辺第 1 項のみ取り出すと

$$\dot{x}(t) = u(t), x(0) = x_0 \quad (10)$$

$$\dot{u}(t) = -\alpha 2F_x(x(t))^T, u(0) = u_0 \quad (11)$$

となる。これは

$$\ddot{x}(t) = -\alpha 2F_{xx}(x(t))\dot{x}(t), x(0) = x_0, \dot{x}(0) = \dot{x}_0 \quad (12)$$

と等価だが、これは摩擦による減衰項のない慣性系であり、消散系である。

(12) の両辺に  $\dot{x}(t)$  をかけて、両辺を時間  $t$  で積分すると

$$\frac{1}{2}\dot{x}(t)^T \dot{x}(t) = -\alpha 2F(x(t)) + C$$

ただしここで  $C$  は積分定数である。初期条件から  $\dot{x}(0) = 0$  であるから、これより  $C = \alpha 2F(x(0))$  を得る。結果として次のようなエネルギー保存則が成り立つ。

$$\frac{1}{2}\dot{x}(t)^T \dot{x}(t) = \alpha 2(F(x(0)) - F(x(t))) \quad (13)$$

この式から次のことがわかる。

(a) 軌道は  $F(x) = F(x(0))$  の等高線から出ることがなく、その等高線より外側における探索は不可能である。

(b) 軌道はカオス的に動き回り収束しない。

項目 (a) は大域的最適解の探索を広い領域で行うためには初期点の位置エネルギーが適当に大きくなければならないことを示唆している。もしくは、等高線  $F(x) = F(x_0)$  より外側の探索をするために初期速度  $u(0) = 0$  (or  $\dot{x}(0) = \dot{x}_0$ ) を与える必要がある。

次にアルゴリズム 1 の (9) の右辺第 2 項のみを取り出すと

$$\dot{x}(t) = u(t), x(0) = x_0 \quad (14)$$

$$\dot{u}(t) = -\alpha F_{xx}(x(t))u(t), u(0) = u_0 \quad (15)$$

となる。これは

$$\ddot{x}(t) = -\alpha F_{xxx}(x(t))\dot{x}(t), x(0) = x_0, \dot{x}(0) = \dot{x}_0 \quad (16)$$

と等価である。ここで力学とのアナロジーにおいて、(16) の右辺は摩擦項に対応することを注意しておく。これは  $F(x)$  が凹である領域では負の摩擦として働き、局所解近傍での  $F(x)$  が凸である領域では正の摩擦として働く。したがって、これは軌道を解へ収束性させることに寄与している。

負のフィードバックである (9) の第 3 項目は安定性に寄与している。

#### 4. カオスの軌道を用いた大域的最適化

アルゴリズム 1 は潜在的には大域的最適解を探す機能を有しているものの、その成功確率は常に高いものではない。

本節ではアルゴリズム 1 を修正することで、大域的最適化のための新しいアルゴリズムを考案する。新しいアルゴリズムを考案するにあたっての要点は以下のようである：

(1) 大域的最適化においては、エネルギー的な考察から、大域的な探索のために尾根を越えて動きまわるためにも高い位置エネルギーを必要とする。一方、局所的あるいは大域的にかかわらず解に収束させるためには、位置エネルギーは低くなければならない。この自己矛盾を解決すつために、2種類の軌道を考えそれぞれに役割を分担させることで、大域的な情報を得ながら、最適解に収束させることを図る。

(2) 大域的最適解と局所最適解の違いは(あくまでも)その評価関数の大小であり、勾配情報のみからはその判別ができない。それゆえ評価関数値そのものを大域的最適化のための手掛りとして有効に取り入れることを試みる。

さて、アルゴリズム 1 を 2種類の軌道を生成する 2つの部分に分割することを考える。一つは高エネルギーのもとで大域的に動き回る”カオスの軌道”と呼ぶ軌道を生成する部分であり、もう一つは低エネルギーのもとで局所的最適解に収束する”収束軌道”と呼ぶ軌道を生成する部分である。

前者には次のような動的システムを用いる。

$$\dot{x}_1(t) = u_1(t), \quad x_1(0) = x_{10} \quad (17)$$

$$\dot{u}_1(t) = -\alpha(2Fx_1(x_1(t))^T), \quad u_1(0) = u_{10} \quad (18)$$

ここで下付添字 1 はカオスの軌道を表し、 $\alpha > 0$  は比例定数である。また、(17) は (9) の右辺第一項に対応している。

また後者には動的システム

$$\dot{x}_2(t) = u_2(t), \quad x_2(0) = x_{20} \quad (19)$$

$$\begin{aligned} \dot{u}_2(t) = & -\alpha F_{x_2} x_2(x_2(t)) u_2(t) \\ & + \delta \frac{(x_1(t) - x_2(t))}{1 + \exp(-\beta(F(x_2(t)) - F(x_1(t))))} \\ & + \gamma u_2(t), \\ u_2(0) = & u_{20} \end{aligned} \quad (20)$$

を用いる。ここで下付添字 2 は収束軌道を表し、 $\alpha > 0, \delta > 0, \beta > 0, \gamma > 0$  は比例定数である。この収束軌道は自ら大域的な探索行為を行わないけれども、カオスの軌道から大域的な情報を受け取れるように考案されている。

(20) の右辺第一項はアルゴリズム 1 の (9) の第二項であり、軌道を局所的に収束させる機能をもっている。

第 3 項は安定化のための負のフィードバック項である。 $u_1(0)$  と  $u_2(0)$  は基本的にゼロであるものとする。

(20) の第二項はシグモイド関数であり、カオスの軌道の評価関数値が収束軌道の評価関数値より小さいときに限り、収束軌道をカオスの軌道の方向に引きつけるような力を与えるように働く。

なお、2 軌道間の情報のやりとりはカオスの軌道から収束軌道への一方向的なものである。

システム (17) ~ (20) は大域的最適化のためにアルゴリズム 1 を改善した、新しく提案するアルゴリズムである。我々はこれをアルゴリズム 2 と呼ぶことにする。

#### 5. シミュレーション

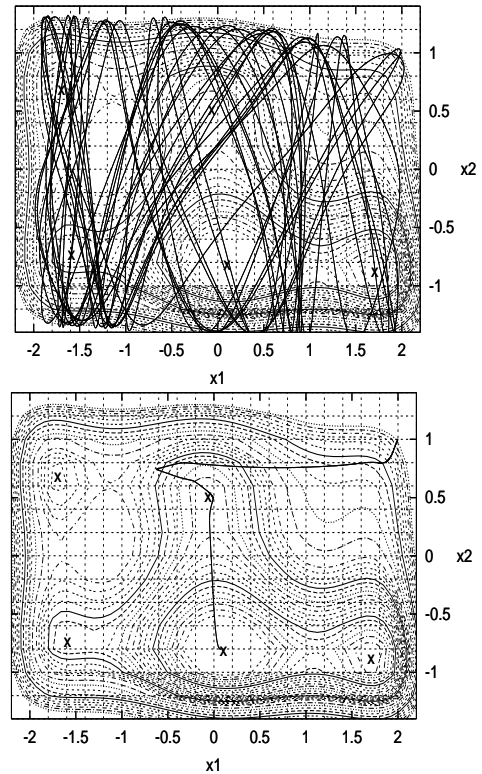


Fig.1 カオスの軌道 (上) と収束軌道 (下)

#### 6. 結論

本論文において、我々は動的数理計画法を互いに干渉しあう 2つのタイプの軌道を生成する 2つの部分に分割した。そして新しく提案したアルゴリズム 2 は高い確率で大域解 (正確にはそのベイシン) に到達することに成功した。

しかしながら、以下のような改善に余地がある。

1. アルゴリズム 2 において、収束軌道は少なくとも大域解のベースインを探し出す能力はもっているが、大域解そのものへ収束するわけではない。この点は改善しなければならない。
2. カオスの軌道は常に (大域的に) 動き回っているわけではない。そこで軌道が大域的に動き回るような工夫が必要である。

これらの項目について、今後の我々の課題としたい。

#### 7. 参考文献

- [S1] K. Shimizu, "Dynamic Nonlinear Programming -New Optimization Algorithms via Dynamic Controllers-", IEEE SMC's99 conference Proceedings.
- [S2] K. Shimizu, K. Suzuki, "A New Global Optimization Algorithm via Dynamic Control," IEEE International Conference, SMC, Oct, 2002.
- [S3] 菅田裕之, 荻野卓朗, 志水清孝, "多軌道型慣性系におけるカオスを用いた大域的最適化手法," 電子情報通信学会論文誌 (A), vol. j79-A No.10 pp.1700-1707, Oct, 1996.