

Nelder-Mead 法および進化型計算によるエネルギープラント運用計画の最適化

1 はじめに

エネルギープラントは工場や建物全体におけるエネルギー消費の大部分を占めており、エネルギー管理者にとってエネルギープラントの省エネルギー化は優先して取り組むべき課題である。エネルギープラントの効率的な省エネルギー手法として、エネルギーマネジメントシステム (EMS) が適用されている。しかしながら EMS を適用するにあたっては、定式化する際のモデルと実際の機器との乖離や、未だ実用的規模のプラントに対する有効な最適化技術がない等、多くの課題がある。現実のエネルギープラントの最適化は困難であるためベンチマーク問題が提案されている。

一方で、生物の進化から着想を得た確率的探索手法である進化型計算 (EC) は、その汎用性の高さから科学や工学の様々な分野で用いられている。

本研究では、実数値最適化問題を解くための有力な EC のアルゴリズムである CMA-ES と多次元非線形最適化問題に適用可能な直接探索法である Nelder-Mead 法を用いて、エネルギープラントの様々な制約条件に柔軟に対応しながら最適化できる手法の構築を目的とする。

2 要素技術

2.1 エネルギープラント問題

エネルギープラントでは、ガスタービン、ボイラ、冷凍機など様々な機器を用いて、電力・熱・蒸気などのエネルギーを供給している。エネルギープラント運用計画問題は、各エネルギーの需給バランス、機器の機械的制約、および運用制約を考慮したうえで、電力購入コストとガス購入コストを最小にするような機器の運転状態を計画する問題である。図 1 に本研究で用いるベンチマーク問題のエネルギープラントの模式図を示す [1]。この問題はガスタービン一台、ボイラ一台、ターボ式冷凍機一台、蒸気吸収式冷凍機二台の 5 つの機器からなる 24 時刻運用問題である。求める目的関数は 24 時刻の各時間帯で購入された電力及びガスの

購入費用であり、

$$\min \sum_{i=1}^I \{C_{E_r}^i E_r^i(x_g^i, x_t^i, \bar{E}_L^i, E_{rm}^i) + C_{F_r}^i(x_g^i + x_b^i)\} \quad (1)$$

と表される。ここで、 $C_{E_r}^i$ は時刻 i における電力購入単価、 $C_{F_r}^i$ は時刻 i におけるガス購入単価、 E_r^i は時刻 i における電力購入量、 \bar{E}_L^i は時刻 i における電力負荷、 E_{rm}^i は時刻 i における電力の残りを表す。 x_g^i は時刻 i におけるガスタービンのガス消費量、 x_t^i は時刻 i におけるターボ式冷凍機の熱生成量、 x_b^i は時刻 i におけるボイラのガス消費量を表す。

また、起動・停止状態は二時間以上継続しなければ状態を変えられないことや動作している機器の出力の上下限制約など制約条件の詳細は、文献 [1] を参照されたい。

2.2 Nelder-Mead 法

Nelder-Mead 法 [2] は制約条件がない最適化問題

$$\min f(x) \quad (2)$$

を解くために用いられる直接探索法である [3]。ここで、 n は次元数、 $f: \mathbb{R}^n \rightarrow \mathbb{R}$ は目的関数を表す。 n 次元の空間で $n+1$ 個の頂点からなる単体 (シンプレックス) を変形・移動する操作をしながら解を探索する。ここで、頂点 x_1, x_2, \dots, x_{n+1} を持つ単体を Δ と表記する。

Nelder-Mead 法は一連の単体を反復的に生成して (2) 式の最適点を近似する。各試行で単体の頂点 $\{x_j\}_{j=1}^{n+1}$ を (3) 式のように目的関数値に従って順序付ける。

$$f(x_1) \geq f(x_2) \geq \dots \geq f(x_{n+1}) \quad (3)$$

ただし x_1 を最良な頂点、 x_{n+1} を最悪の頂点とする。

アルゴリズムで用いる単体の操作には反射、拡大、収縮、縮小があり、それぞれにスカラー値のパラメーター α (反射)、 β (拡大)、 γ (収縮)、 δ (縮小) が与えられている。ただし、パラメータの値は、 $\alpha > 0$ 、 $\beta > 1$ 、 $0 < \gamma < 1$ 、 $0 < \delta < 1$ を満たす。Nelder-Mead 法の標準的な実装ではパラメーターは (4) 式のように選ばれている。

$$\{\alpha, \beta, \gamma, \delta\} = \{1, 2, 1/2, 1/2\} \quad (4)$$

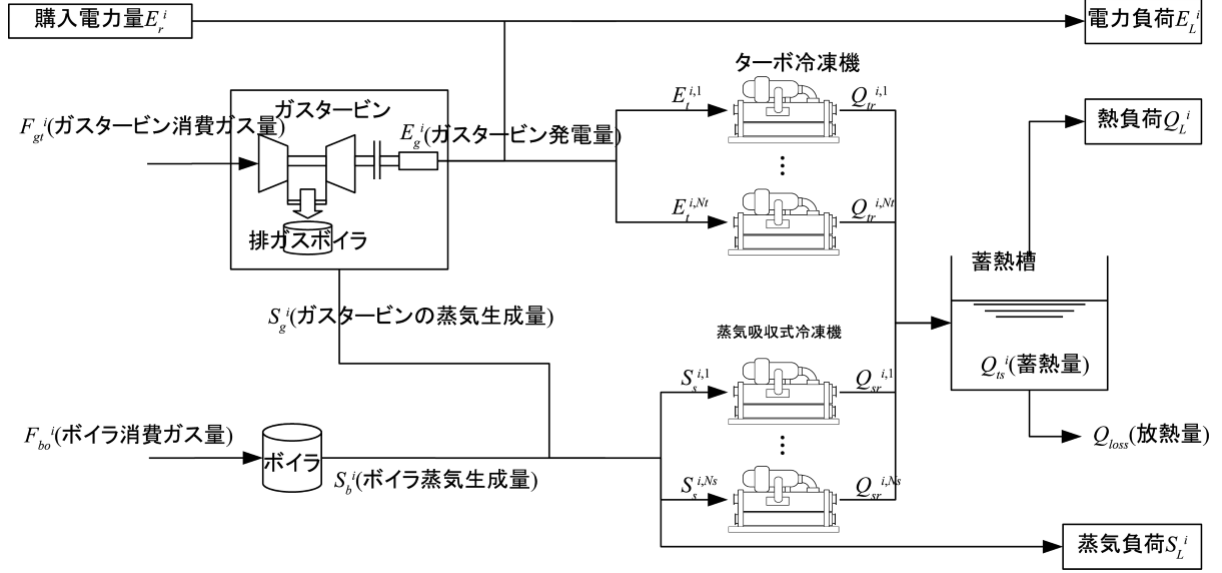


図 1: エネルギープラントの模式図 (文献 [1] の図 3.2.1 参照)

\bar{x} を、目的関数が小さい順で n 番目までの頂点の重心とすると、(5) 式で表される。

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i \quad (5)$$

ここで、Nelder-Mead 法での 1 試行のアルゴリズムの概要を述べる。

1. ソート: Δ の $n+1$ 個の頂点を f で評価し、(3) 式が成立するようにソートする。
2. 反射: 反射点 x_r を (6) 式で計算する。

$$x_r = \bar{x} + \alpha(\bar{x} - x_{n+1}) \quad (6)$$

$f_r = f(x_r)$ を評価する。もし $f_1 \leq f_r < f_n$ であれば、 x_{n+1} を x_r に置き換える。

3. 拡大: もし $f_r < f_1$ であれば、(7) 式で拡大点 x_e を計算し、 $f_e = f(x_e)$ を評価する。

$$x_e = \bar{x} + \beta(x_r - \bar{x}) \quad (7)$$

$f_e < f_r$ のとき、 x_{n+1} を x_e に置き換える。そうでなければ、 x_{n+1} を x_r に置き換える。

4. 外側の収縮: もし $f_n \leq f_r < f_{n+1}$ であれば、(8) 式で外側の収縮点 x_{oc} を計算し、 $f_{oc} = f(x_{oc})$ を評価する。

$$x_{oc} = \bar{x} + \gamma(x_r - \bar{x}) \quad (8)$$

$f_{oc} \leq f_r$ のとき、 x_{n+1} を x_{oc} に置き換える。そうでなければ、Step 6 に移行する。

5. 内側の収縮: もし $f_r \geq f_{n+1}$ であれば、(9) 式で内側の収縮点 x_{ic} を計算し、 $f_{ic} = f(x_{ic})$ を評価する。

$$x_{ic} = \bar{x} - \gamma(x_r - \bar{x}) \quad (9)$$

$f_{ic} < f_{n+1}$ のとき、 x_{n+1} を x_{ic} に置き換える。そうでなければ、Step 6 に移行する。

6. 縮小: $2 \leq i \leq n+1$ に対し、(10) 式を計算する。

$$x_i = x_1 + \delta(x_i - x_1) \quad (10)$$

2.3 CMA-ES

進化型計算のひとつである共分散行列進化適応戦略 (Evolution Strategy with Covariance Matrix Adaptation, CMA-ES)[4] は悪スケール性や変数間依存といった最適化を困難にする特徴を持つ実数値最適化問題に対する有力なアルゴリズムである [5]。CMA-ES による探索では個体は多変量正規分布を用いた突然変異により生成される。この多変量生成分布の統計量である平均ベクトルおよび共分散行列を更新していくことで、探索を進める。

3 問題設定

ガスタービン一台、ボイラ一台、ターボ式冷凍機一台、蒸気吸収式冷凍機二台の 5 つの機器からなる 24 時刻運用問題であり、120 次元の入力変数 x が存在する。

表 1 に x の定義域を示す．なお，変数の定義域は各機器が稼働時のものであり，停止時は 0 となる．

表 1: 変数説明

変数	変数の定義域	変数の意味
x_t	1.5 ~ 5.0	ターボ式冷凍機の熱生成量
x_{s1}	4.5 ~ 15.0	蒸気吸収式冷凍機 1 の熱生成量
x_{s2}	4.5 ~ 15.0	蒸気吸収式冷凍機 2 の熱生成量
x_g	1103 ~ 3679	ガスタービンのガス消費量
x_b	8.02 ~ 803	ボイラのガス消費量

4 提案手法

今回は Nelder-Mead 法と CMA-ES を用いた電力プラントの最適化の第一段階として，エネルギープラント運用計画のための最適化ベンチマーク問題 ver.1[1] を対象とする．なお，deap の cma の Strategy を使用した [6]．

単目的最適化の場合 (Strategy) は (1) 式で述べた元の目的関数を f ，機器の上下限制約，起動・停止状態などの満たすべき等式・不等式をまとめた制約違反関数を V ，ペナルティ関数の重みを ρ とすると，Nelder-Mead 法及び CMA-ES 上での目的関数 F は (11) 式で表される．

$$F = f + \rho V \quad (11)$$

今回の実験では，制約違反関数 V が 1.0×10^{-10} より小さい解を実行可能解と定義する．

予備実験により，Nelder-Mead 法はそのまま適用しても F を十分に小さくできないことがわかったため，決定変数の一部である x を固定し，2 値で各機器の稼働状態を表す y のみの探索をした． $x_i (i = 0, 1, \dots, 120)$ は $y_i = 0$ ，すなわち機器が停止状態のときは 0，そうでないときは各上下限値の平均値とした．そして，Nelder-Mead 法で見つかった最後の解を CMA-ES の初期平均ベクトルとし，初期個体群に加えた．

Nelder-Mead 法では各値が離散値である y を連続変数で探索できるように，最初の単体における頂点座標の各値を 0.99 以上 1.01 以下のランダムな実数と設定して探索し，目的関数内で各値が 1 より小さければ 0，そうでないとき 1，と処理した後に評価した．

CMA-ES では f が定めた閾値 t より小さくなったから F にペナルティを与えることで， ρ を小さくしても探索中に目的関数が小さくなりすぎて実行不可能な領域を探索することを防いだ．

5 実験

表 2 に Nelder-Mead 法で用いた実験パラメータを示す．表 3 に CMA-ES で用いた実験パラメータを示す．閾値 t は，Nelder-Mead 法により $\{f, V\} = \{3999381.510, 99.58317874\}$ が得られ，実行可能解の f がこの付近の値となると予測したことから決定した．

表 2: Nelder-Mead 法の実験パラメータ

パラメータ	値
変化とみなす最小変化量	1.0×10^{-5}
変化しない場合終了するステップ数	200
最大試行数	1000
ρ (ペナルティ関数の係数)	1.0×10^3

表 3: CMA-ES の実験パラメータ

パラメータ	値
σ (初期標準偏差)	0.1
世代数	5000
入力変数の次元	120
一世代の個体数 λ	1200
ρ (ペナルティ関数の係数)	1.0×10^4
t (閾値)	3999000

表 4 に実験で得た最終的な目的関数値 f ，制約違反関数 V および既知解を示す．結果として既知解 3 より

表 4: 実験結果

解法	目的関数値	制約違反関数
既知解 1	3999631.278	2.44×10^{-10}
既知解 2	3999635.845	6.43×10^{-12}
既知解 3	4052185.662	3.93×10^{-14}
実験	4033408.653	9.78×10^{-11}

目的関数値が小さい実行可能解を得ることができた．

次に，図 2 に最適化をした目的関数 F の推移を，図 3 に元の目的関数 f の推移を，図 4 に制約違反関数 V の推移を示す．縦軸はそれぞれ F, f, V ，横軸は Nelder-Mead 法の試行数と CMA-ES の世代数である．ペナルティ関数の係数 ρ が 2 手法間で異なり，Nelder-Mead 法は 430 試行であったため，その前後で F が急に増加している． f は Nelder-Mead 法で 400 万ほどまで小さ

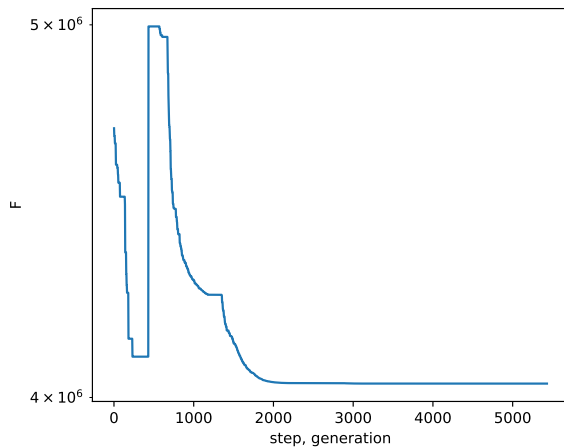


図 2: 目的関数値 F の推移

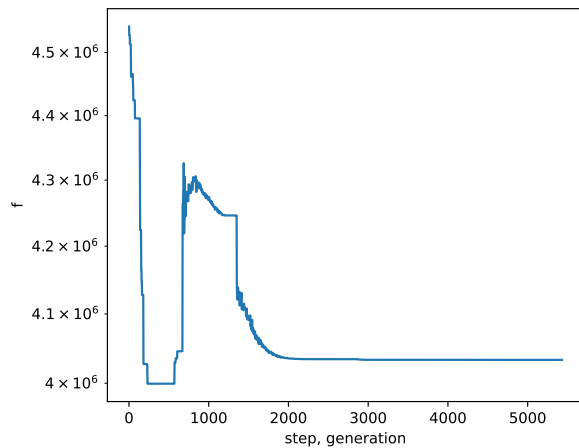


図 3: 目的関数値 f の推移

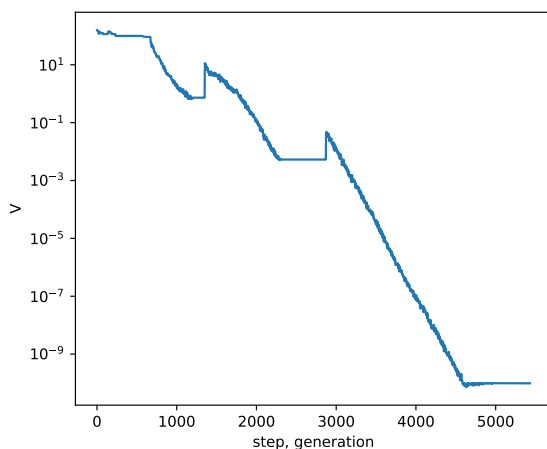


図 4: 制約違反関数 V の推移

くできたが、その後の CMA-ES で一度 430 万程度まで増加した。また、そのときの V は減少傾向にある。これは ρ が大きく、 V を小さくすることを優先したためだと考えられる。Nelder-Mead 法では、 V は最初から低く、あまり減少していない。これは x を適切な値に固定したためだと考えた。

6 まとめと今後の課題

本研究では Nelder-Mead 法と CMA-ES による電力プラントの最適化について検討し、既知解に近い実行可能解を得ることができたが、最良の既知解を改善することはできなかった。今後の課題として、ペナルティ関数の係数を最適化の途中で調整することや、各手法の実験パラメータの調整があげられる。

参考文献

- [1] 電気学会情報知能システムの新展開とその産業応用調査専門委員会. 産業応用のための最適化ベンチマーク問題集. 電気学会技術報告 第 1287 号, 3.2 節, 2013.
- [2] J. A. Nelder and R. Mead. A Simplex Method for Function Minimization. *The Computer Journal*, Vol. 7, No. 4, pp. 308–313, 01 1965.
- [3] Fuchang Gao and Lixing Han. Implementing the nelder-mead simplex algorithm with adaptive parameters. *Computational Optimization and Applications*, Vol. 51, No. 1, pp. 259–277, 2012.
- [4] N. Hansen and A. Ostermeier. Adapting arbitrary normal mutation distributions in evolution strategies: the covariance matrix adaptation. In *Proceedings of IEEE International Conference on Evolutionary Computation*, pp. 312–317, 1996.
- [5] 秋本洋平. Evolution strategies による連続最適化. システム / 制御 / 情報, Vol. 60, No. 7, pp. 292–297, 2016.
- [6] Félix-Antoine Fortin, François-Michel De Rainville, Marc-André Gardner, Marc Parizeau, and Christian Gagné. DEAP: Evolutionary algorithms made easy. *Journal of Machine Learning Research*, Vol. 13, pp. 2171–2175, jul 2012.