

ベイズ最適化・メッシュモーフィングを用いた抗力最小化

オープンソースを用いた形状最適化

片山 達也 [1†]

[1] オープン CAE 勉強会@関西

Drag minimization using Bayesian optimization and Mesh morphing

Shape optimization using open source

TATSUYA KATAYAMA [*†]

[*] OpenCAE Study Meeting@Kansai

Abstract

A Bayesian optimization method is used for optimizing hyperparameters in neural networks. A Bayesian optimization can efficiently solve multimodal problems by evaluating uncertainty using Gaussian process regression. Apply Bayesian optimization and mesh morphing to the drag minimization problem and examine the penguin swimming posture.

Keywords: Bayesian optimization, Mesh morphing, OpenFOAM®

1. はじめに

近年ディープラーニングになど機械学習が急速な進化及び技術のオープン化(オープンソース化)を遂げているが，それを支える周辺技術も見直され発展してきている．学習計算コストの高いディープラーニングにおいて，ニューラルネットワークのハイパーパラメータの最適化に用いられているベイズ最適化[1]もその一つである．本稿ではオープンソースの恩恵を賜り，古くも新しい技術であるベイズ最適化を CFD の形状最適化について検討を行う．

2. ベイズ最適化

2.1. 概説

ベイズ最適化は，ガウス過程により平均と分散を考慮した回帰式について，最小（最大）となりうる可能性のある部分に実験点を追加し，回帰式を更新するとともに最小値（最大値）を探す手法である．図 1 は未知の 1 変数関数最小化問題におけるベイズ最適化の様子である．

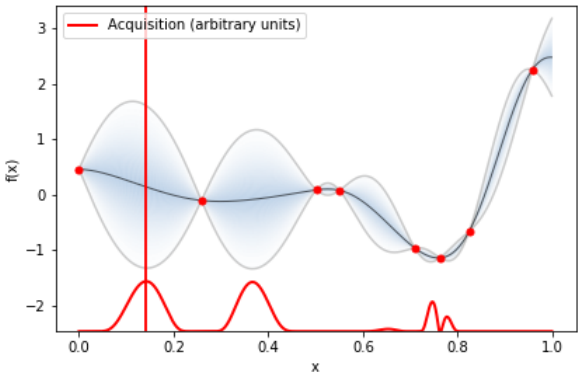

Fig. 1 Bayesian Optimization of 1D black-box function

† *E-mail address of corresponding author: katayama1227ron@gmail.com*

図中の黒色線が，赤色点で表される実験点をもとにガウス過程回帰した回帰式である．また灰色線内の青色グラデーション区間が回帰式の信頼区間である．図 1 において実験点はすでに 8 点存在し，回帰式と実験点を見る限り最小解は x = 0.75 付近と考えられる．一方，回帰式の不確かさを表す信頼区間を考慮すると x = 0.15 付近でまだ最小値となりうる可能性がある．ベイズ最適化では最小となりうる可能性のある x = 0.15 付近を次の実験点とし，不確かさを加味して最小値を探す．

以上からわかるようにベイズ最適化は効率的な実験を計画できる多峰性にも強い最適化手法である．

2.2. GPyOpt

GPyOpt [2]は Sheffield 大学の Machine Leaning グループが開発したベイズ最適化用の Python オープンソースライブラリである．またガウス過程のライブラリには同グループ開発の GPy を用いている．付録 A に示すわずか数行の Code でベイズ最適化が実施できる見通しの良いクラス設計がなされている．

3. メッシュモーフィング

3.1. 概説

メッシュモーフィングの一つである自由形状変形(Free-Form Deformation) [3, 4]は，物体の周りに単純立方格子に写像可能な格子点を作成し，単純立方格子として写像した後，ベジエ立体として変形させ物理空間に逆写像させる手法である．最適化を行うための設計パラメータとして自由形状変形を利用する

3.2. PyGEM

PyGEM [5]は SISSA mathLab(先端研究国際大学院大学(イタリア))で開発されている Python のオープンソースメッシュモーフィングライブラリである．対応フォーマット多種(iges, step, stl, unv, OpenFOAM, vtk, LS-Dyna)あり，放射基底関数法(RBF)や逆距離加重法(IDW)にも対応している．図 2 に OpenFOAM のメッシュモーフィング例を示す．

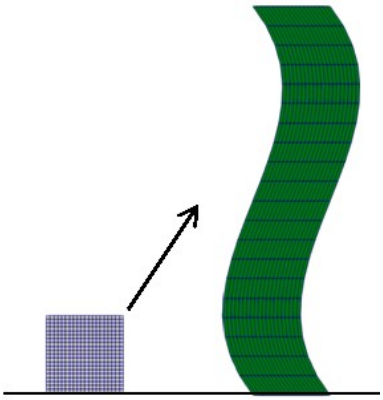

Fig. 2 Cavity Mesh in OpenFOAM to Dried squid [6]

4. ペンギンの抗力最小となる遊泳姿勢

2，3 章で概説した 2 つライブラリを用いて，今後ペンギンの抗力最小となる遊泳姿勢を提案する.(図 3)

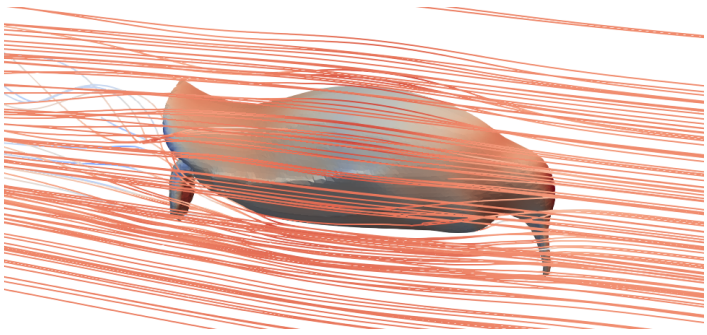

Fig. 3 Before optimization and Penguin swims by looking down

5. おわりに

オープンソースを用いた形状最適化に必要な技術を概説した.

参考文献

[1] Jonas Mockus, Vytautas Tiesis, and Antanas Zilinskas. The application of Bayesian methods for seeking the extremum. Towards Global Optimization, 2:117–129, 1978.

[2] Machine Learning group of the University of Sheffield. GPyOpt, 2018. http://sheffieldml.github.io/GPyOpt/, (accessed 2018-11-18).

[3] Sederberg TW, Parry SR. Free-form deformation of solid geometric models. In: ACM SIGGRAPH computer graphics, vol. 20. New York: ACM; 1986. p. 151–60. http://faculty.cs.tamu.edu/schaefer/teaching/689_Fall2006/p151-sederberg.pdf, (accessed 2018-11-19)

[4] Tezzele et al .Dimension reduction in heterogeneous parametric spaces with application to naval engineering shape design problems. Adv. Model. and Simul. in Eng. Sci, 2018. https://link.springer.com/article/10.1186/s40323-018-0118-3, (accessed 2018-11-19).

[5] SISSA mathLab, PyGEM, GitHub, 2016. https://github.com/mathLab/PyGeM, (accessed 2018-11-19)

[6] TatsuyaKatayama OpenFOAM の Cavity チュートリアルのメッシュをワカメにする Qiita 2017 https://qiita.com/TatsuyaKatayama/items/d1fef4e6ea5a3053cff8, (accessed 2018-11-19).

[7] Machine Learning group of the University of Sheffield. First steps, GPyOpt, 2018. http://sheffieldml.github.io/GPyOpt/firstexamples/index.html, (accessed 2018-11-19).

付録 A　GPyOpt でのベイズ最適化例

GPyOpt の First steps [7]として紹介されている GPyOpt の使用例を Code 1 に示す.

Code 1　Header of template_OpenCAE_symposium.tex.

```
%# --- Load GPyOpt
from GPyOpt.methods import BayesianOptimization
import numpy as np

# --- Define your problem
def f(x): return (6*x-2)**2*np.sin(12*x-4)
domain = [{'name': 'var_1', 'type': 'continuous', 'domain': (0,1)}]

# --- Solve your problem
myBopt = BayesianOptimization(f=f, domain=domain)
myBopt.run_optimization(max_iter=15)
myBopt.plot_acquisition()
```