

Bayesian Network のプログラム作成
中村 康昭

1 今月の課題

Bayesian Network を用いた最適化プログラムの作成し、動作確認を行う。また、ビット間に依存関係がある問題に対して Bayesian Network を用いることで依存関係を抽出ができることを確認する。

2 研究の進捗状況

現在、The Bayesian Optimization Algorithm(BOA) のプログラムの作成し、検証を行っている。

2.1 対象問題

対象問題としては、Goldberg ら²⁾と同様の物を用いる。ここでは、用いられている関数の一つである次数 3 のだまし問題 (deceptive function of order 3:3deceptive) について示す。これは、式 (1)、および式 (2) によって定義される。

$$f(X) = \sum_{i=0}^{l-1} f_k(S_i) \quad (1)$$

$$f_{3deceptive}(u) = \begin{cases} 0.9 & \text{if } u=0 \\ 0.8 & \text{if } u=1 \\ 0 & \text{if } u=2 \\ 1.0 & \text{if } u=3 \end{cases} \quad (2)$$

式 (1) において、 S_i は次数 (今回は 3) の中における 1 のビット数を示している。また、 f_k は関数の種類を示し、 l は設計変数の数を示す。

2.2 パラメータ設定

3deceptive を対象問題とする数値実験に際し、用いたパラメータを Table 1 に示す。

Table 1 用いたパラメータ

個体数	100
優良個体数	50
設計変数長	3
遺伝子長	30
最大入力矢印本数	2
終了世代	100
ネットワークの信頼指標	K2 metric

2.3 実験結果

2.1 節で述べた関数に対して、Bayesian Network を構築し、ビット間に存在する依存関係を認識しているかどうかを検討した。今回は設計変数を 10 として実験を行ったが、ここでは 0~3 ビット目に着目し、依存関係を示す矢印がどのビットからどのビットへ入るかを確認した。Table 2 に示される数字は 100 世代終了後に、20 回試行中、何度そのビット間に依存関係の矢印が示されたかである。

Table 2 実験結果

	node0	node1	node2
node1	18	-	-
node2	20	20	-
node3	11	13	12

Table 2 において、列はどのビットから矢印が出ているかを示し、行はどのビットに矢印が入るかを示す。つまり、0~2 ビット目までには依存の関係があり、3 ビット目には依存関係があまりないように認識されていることが確認できる。

3 翌月への課題

様々な問題に適用してデータを収集し、引き続きプログラムの検証を行うことが今後の課題となる。また、ビット長がのびたときに sGA であれば指数関数的である評価計算回数の増加が、BOA を用いることにより扇形になるという報告がある²⁾ため、こちらについても調査を行う。

参考文献

- 1) Dan Geiger, David M. Chickering.
Learning Bayesian Networks: The Combination of Knowledge and Statistical Data
(Microsoft Technical Report MSR-TR-94-09,1994)
- 2) Martin Pelikan, David E. Goldberg, and Erick Cantu-Paz.
BOA: The Bayesian Optimization Algorithm
(IlligAL Report No.99003, 1999)