

Black-box 最適化に対する Budget を考慮した探索空間の初期化

Initialization of a Search Space Considering a Budget for Black-box Optimization

野村 将寛 阿部 拳之
Masahiro Nomura Kenshi Abe

株式会社サイバーエージェント
CyberAgent, Inc.

Black-box optimization is a problem of optimizing the objective function within the bounds of a given budget for evaluations. In Black-box optimization, it is generally assumed that the calculation cost for evaluating one solution is large, so it is important to search efficiently with as few budgets as possible. However, there is a problem that state-of-the-art black-box optimization methods such as Bayesian optimization and CMA-ES do not consider a budget. In this paper, we aim to propose an initialization of a search space that takes a budget into account, dealing with the above problem. We confirm that the proposed method shows good performance by experiments on the benchmark functions.

1. はじめに

Black-box 最適化は、与えられた評価用 budget B の制限内で目的関数 $f: \mathcal{X} \mapsto \mathbb{R}$ を最適化する問題である。つまり、 B 回の関数評価が可能な前提で、可能な限り低い評価値 $f(\mathbf{x})$ を持つ解 $\mathbf{x} \in \mathcal{X}$ を得ることが目的となる。Black-box 最適化において、 f の代数的表現は与えられず、勾配情報は利用できない。

Black-box 最適化では、一般に 1 回の解の評価にかかる計算コストが大きいことを想定しているため、できるだけ少ない budget で効率よく探索を行うことが重要となる。Black-box 最適化における応用例として、機械学習手法のハイパーパラメータ最適化 [Bergstra 11, Ilievski 17, Golovin 17], エージェントベースシミュレーションのパラメータ調整 [Yang 09], 飛行機的设计問題 [Chiba 05] などが挙げられる。深層生成モデルの 1 つである Generative adversarial networks (GAN) [Goodfellow 14] の性能は、ハイパーパラメータの選択に大きく依存していることが文献 [Lucic 18] により指摘されている。ここで、GAN は 1 回の学習に長時間必要なため、ハイパーパラメータ最適化のために用いることができる budget は制限されることが考えられる。このように、与えられた budget に応じて探索と活用のトレードオフをとることが重要である。

しかし、ベイズ最適化 [Brochu 10] や CMA-ES [Hansen 03] をはじめとする state-of-the-art な Black-box 最適化手法は budget を考慮していないという問題点が存在する。ベイズ最適化は探索空間全体に対する代理関数から獲得関数を構築し、それを最適化することで次の探索点を得るアルゴリズムである。budget が非常に小さいとき、正確な代理関数のモデル化 (探索) に budget を消費してしまうため、評価値の良い領域に対する最適化 (活用) が不十分になってしまうことが考えられる。実用上、探索空間が高次元である場合や探索空間の範囲を広く定義している場合に budget が不十分になる傾向があると考えられる。

本論文では、上記の問題点に対処した budget を考慮した探索空間の初期化法の提案を目的とする。提案手法が優れた性能を示すことをベンチマーク関数に対する実験により確認する。

連絡先: 野村 将寛, 株式会社サイバーエージェント, Email: nomura_masahiro@cyberagent.co.jp

2. 提案手法

探索空間としてある任意の超立方体 $\mathcal{X} \subseteq \mathbb{R}^d$ (d : 次元数) が与えられたとき、提案手法は階層的分割により探索空間の絞り込みを行うことで、有望であると考えられる領域 $\mathcal{X}_S \subseteq \mathcal{X}$ を出力する。これにより、ベイズ最適化などの最適化手法は \mathcal{X} の代わりに \mathcal{X}_S を初期探索範囲とした上で最適化を実行することが可能になる。

2.1 節で提案手法を用いた最適化フレームワーク, 2.2 節で探索空間の絞り込みに用いる budget の計算, 2.3 節で探索空間の絞り込み方法の詳細, 2.4 節で提案手法の理論解析を述べる。

2.1 最適化フレームワーク

提案手法により探索空間を初期化する最適化フレームワークを Algorithm 1 に示す。1 行目で全体の budget B から探索空間の絞り込みに用いる budget B_{ref} の計算, 2 行目で有望な探索空間の絞り込み, 3 行目で絞り込まれた探索空間を初期探索範囲とした最適化を実行している。

Algorithm 1 提案手法により探索空間を初期化する最適化フレームワーク

Require: budget B , search space \mathcal{X} , dimension number d , γ , optimizer

- 1: $B_{\text{ref}} \leftarrow \gamma \cdot B$
- 2: $\mathcal{X}_S \leftarrow \text{refine_search_space}(d, \mathcal{X}, B_{\text{ref}})$
- 3: $\mathbf{x}_{\text{best}} \leftarrow \text{optimize with a given optimizer for the search space } \mathcal{X}_S \text{ until a budget reach } B$

次元数 $d = 2$, budget $B = 8$, 最適化手法を Random Search とした場合の提案手法の概念図を図 1 に示す。右図では、提案手法により絞り込みを行った探索空間に対して、Random Search を実行している。

2.2 探索空間の絞り込みに用いる budget の計算

全体の budget B に対して、提案手法に使用する budget B_{ref} を $B_{\text{ref}} = \gamma \cdot B$ と設定する。ここで、次元数 $d (\geq 1)$ に対して γ は $\gamma = 0.59 \cdot \exp(-0.033B/d)$ により計算を行う。 γ は $B \rightarrow \infty$ のとき $\gamma \rightarrow 0$ となる。

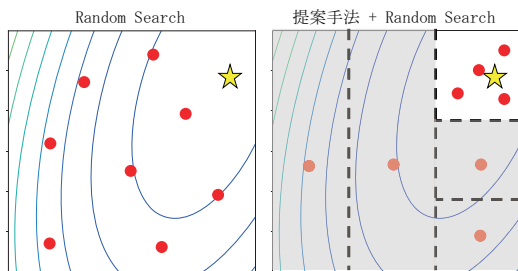


図 1: 次元数 $d = 2$, budget $B = 8$, 最適化手法を Random Search とした場合の提案手法の概念図. 左図は Random Search のみ, 右図は提案手法 + Random Search を示す. 右図の灰色の領域は, 絞り込みを行った結果除外された領域を表す.

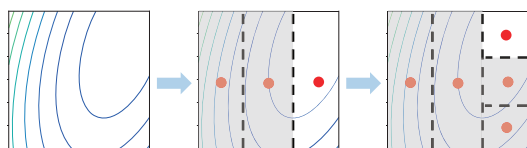


図 2: 次元数 $d = 2$, 分割数 $K = 3$ の場合の提案手法の絞り込み方法. ある次元について K 分割し, 中心点について評価を行う. 中心点在最良となった領域を残し, 別の次元に対して再び K 分割を行うことを繰り返す.

2.3 探索空間の絞り込み方法

2.3.1 アルゴリズム

提案手法は各次元に対して領域を等間隔で分割することにより, 有望領域の絞り込みを行う. 次元数 $d = 2$, 分割数 $K = 3$ の場合の提案手法の絞り込みを図 2 に示す. 提案手法では, ランダムに選択したある次元に対して K 分割し, 分割された領域の中心点の評価値が最も良い領域のみを残す. この操作を, 全ての次元について分割が終了するまで繰り返す.

提案手法による探索空間の絞り込みのアルゴリズムを Algorithm 2 に示す.

Algorithm 2 refine_search_space($d, \mathcal{X}, B_{\text{ref}}$)

Require: dimension number d , search space \mathcal{X} , budget for refinement B_{ref}

- 1: $K = \arg \max_{k \in \mathbb{N} \setminus 2\mathbb{Z}} \{B_k : B_k \leq B_{\text{ref}}\}$
 - 2: **if** $K \leq 1$ **then**
 - 3: return \mathcal{X}
 - 4: **end if**
 - 5: initialization : $\mathcal{X}_S = \mathcal{X}$
 - 6: **for** $i = 1$ to d **do do**
 - 7: randomly select d_i without replacement from index set of dimensions
 - 8: divide \mathcal{X}_S into $\{X_k\}_{k=1}^K$ with respect to dimension d_i
 - 9: $k^* \leftarrow \arg \min_{k \in [K]} f(x_k)$, x_k is a center point of X_k
 - 10: update the search space : $\mathcal{X}_S \leftarrow X_{k^*}$
 - 11: **end for**
 - 12: return \mathcal{X}_S
-

2.3.2 分割数の計算

分割数 K が奇数である場合に, 分割に必要な budget は $B_K = K + \sum_{l=1}^{d-1} (K-1)$ となる. 提案手法では, 式 (1) により B_K が B_{ref} に最も近づくように奇数の分割数 $K \in \mathbb{N} \setminus 2\mathbb{Z}$ を決定する.

$$K = \arg \max_{k \in \mathbb{N} \setminus 2\mathbb{Z}} \{B_k : B_k \leq B_{\text{ref}}\}. \quad (1)$$

2.4 理論解析

提案手法により得られた新たな探索空間を $\mathcal{X}_S \subseteq \mathcal{X} (\subseteq \mathbb{R}^d)$ とする. ここで, \mathcal{X}_S についての損失を表す 2 種類の指標 $r_{\text{inf}}(\mathcal{X}_S)$ と $r_{\text{sup}}(\mathcal{X}_S)$ をそれぞれ式 (2), (3) で定義する.

$$r_{\text{inf}}(\mathcal{X}_S) = \inf_{\mathbf{x}_S \in \mathcal{X}_S} f(\mathbf{x}_S) - \inf_{\mathbf{x} \in \mathcal{X}} f(\mathbf{x}), \quad (2)$$

$$r_{\text{sup}}(\mathcal{X}_S) = \sup_{\mathbf{x}_S \in \mathcal{X}_S} f(\mathbf{x}_S) - \inf_{\mathbf{x} \in \mathcal{X}} f(\mathbf{x}). \quad (3)$$

以下では $d = 1$ の場合について定理と系を説明するが, $d \geq 2$ についても, 目的関数が変数分離可能 ($f(\mathbf{x}) = \sum_{i=1}^d f_i(x_i)$) である場合はそのまま成立する.

定理 1. 元の探索空間 \mathcal{X} がコンパクト, f が厳密な準凸関数であり, $x = a$ ($a \in \mathbb{R}$) で対称であるとする. このとき, $\forall K \in \mathbb{N} \setminus 2\mathbb{Z}$ に対して $r_{\text{inf}}(\mathcal{X}_S) = 0$.

Proof. 仮定より, $x \geq a$ のとき, f は x に対して単調増加, $x < a$ のとき, f は x に対して単調減少となる. K 分割された探索空間を $\{X_i\}_{i=1}^K$ とする. つまり, $\mathcal{X} = \bigcup_{i=1}^K X_i$ であり, 各 X_i は互いに素である. ここで, 提案手法が出力した領域 X_k でない領域 X_j ($j \in [K] \setminus \{k\}$) に最適解 $x^* = \arg \min_{x \in \mathcal{X}} f(x)$ が存在すると仮定する. つまり, $x^* \in X_j$ ($j \in [K]$), $f(x_k) \leq f(x_i)$ ($k \neq j, \forall i \in [K] \setminus \{k\}$) と仮定する. また, 各 $X_i, i \in [K]$ の長さを l とし, $x_i, i \in [K]$ を X_i の中心点とする.

- (i) $a \in \mathcal{X}$ のとき: $x^* (= a) \in X_j$ より, x_j は x^* から最大 $l/2$ しか離れていない. よって, x に対する f の単調性と $x = a$ での対称性より, $f(x_j) \leq f(x^* + l/2) = f(x^* - l/2)$. ここで, $x_k \in X_k$ に対して上記と同様に, $f(x_k) > f(x^* + l/2) = f(x^* - l/2) = f(x_j)$. これは $f(x_k) \leq f(x_i)$ に矛盾する.
- (ii) $a \notin \mathcal{X}$ のとき: まず, $x_j > a$ の場合を考える. このとき, $x^* = x_j - l/2$ となるため, x に対する f の単調性より, $f(x_j) = f(x^* + l/2)$. ここで, $x_k \in X_k$ に対して上記と同様に, $f(x_k) > f(x^* + l/2) = f(x_j)$. これは $f(x_k) \leq f(x_i)$ に矛盾する. $x_j \leq a$ の場合も, $x = a$ における f の対称性より, 同様に示される.

よって, 提案手法が出力した領域に最適解 x^* が存在するため, $r_{\text{inf}}(\mathcal{X}_S) = 0$. \square

系 1. 元の探索空間 \mathcal{X} がコンパクト, f が厳密な準凸関数であり, $x = a$ ($a \in \mathbb{R}$) で対称であるとする. ここで \mathcal{X} の長さを M とすると, $\sup_{x \in \mathcal{X}_S} f(x) \leq f(a - \frac{M}{K}) = f(a + \frac{M}{K})$.

定理 2. 元の探索空間 \mathcal{X} がコンパクトであり, f が \mathcal{X} で連続であるとする. 探索空間の絞り込みに用いる budget $B_{\text{ref}} \rightarrow \infty$ のとき, 損失 $r_{\text{sup}}(\mathcal{X}_S) \rightarrow 0$ となる.

Proof. 複数の大域的最適解を持つ場合のため、大域的最適解集合 $\arg \min_{x \in \mathcal{X}} f(x)$ それぞれの要素を含む領域の集合族を \mathbb{X}^* , 最適な領域に対応するインデックス集合を $I^* \subseteq [K]$ と定義する. まず, $B_{\text{ref}} \rightarrow \infty$ のとき $K \rightarrow \infty$ である. また, \mathcal{X} のコンパクト性より, \mathcal{X} の長さを M とすると, 分割数 K のとき領域 X_i ($i \in [K]$) の長さは $l^{(K)} = M/K$ となる. すると, $\forall X_j \in \mathbb{X}^*, \forall x \in X_j, x^* \in X_j$ に対し, $|x - x^*| \leq l^{(K)}$. よって, $K \rightarrow \infty$ のとき, $x \rightarrow x^*$. ここで, f は \mathcal{X} で連続より, $\forall x \in X_j$ について $\lim_{x \rightarrow x^*} f(x) = f(x^*)$. 一方, $i \in [K] \setminus I^*$ について, 領域 X_i は x^* を含まないので, $\inf_{x \in X_i} f(x) > f(x^*)$. このとき, 提案手法は \mathcal{X}_S としていずれかの $X_j \in \mathbb{X}^*$ を出力するので, $\lim_{K \rightarrow \infty} \sup_{x \in \mathcal{X}_S} f(x) = \lim_{x \rightarrow x^*} \sup_{x \in X_j} f(x) = f(x^*)$. \square

3. 実験

3.1 目的

本実験では, budget が制限された Black-box 最適化に対する提案手法の優位性の確認を目的とする.

3.2 比較手法

state-of-the-art なベイズ最適化手法の1つである Tree-structured Parzen Estimator (TPE) [Bergstra 11] を用いた上で, 提案手法による初期化を用いなかった場合と用いた場合を比較する. 本実験では, TPE のみを用いた場合 (提案手法による初期化を用いなかった場合), 提案手法による初期化を用いた場合をそれぞれ TPE, Refine+TPE と表記する. TPE の実装には Hyperopt^{*1} を用いた.

3.3 ベンチマーク関数

本実験で使用する6つのベンチマーク関数の名称, 定義式, 次元, 探索範囲を表1に示す. ここで, Shekel, Hartmann 関数に現れる係数は文献 [Surjanovic 14] から引用している.

3.4 性能評価指標

本実験では, budget を $B = 10 \times d$ とした上で, 50 試行における最良評価値の平均と標準偏差の推移を性能評価指標として用いる.

3.5 結果

TPE と Refine+TPE をベンチマーク関数に適用したときの50 試行における最良評価値の平均と標準偏差の推移を図3に示す. Refine+TPE は全てのベンチマーク関数において, TPE より良い性能を示している.

3.6 考察

本節では, 提案手法による初期化がアルゴリズムの振る舞いに与える影響について考察を行う. 図4に, Hartmann 関数に対する TPE と Refine+TPE の典型的な挙動を示す. 横軸は評価回数, 縦軸は各評価時に得られた評価値を表す. TPE が探索終盤でも評価値の良い領域に解をサンプルできていないのに対し, Refine+TPE では有望領域の絞り込み後には評価値の良い解が多数サンプルされていることがわかる. これより, 提案手法による初期化が budget の制限された問題に対し有効に機能することが示唆される.

4. おわりに

本論文では, Black-box 最適化に対する budget を考慮した初期化法の提案を行った. 提案手法は budget に応じて探索空

間を分割することで, 超立方体を探索範囲とする最適化手法に対して有望な領域を初期探索空間として与えることが可能である. ベンチマーク関数において, 提案手法による初期化を用いた場合はそれを用いない場合より良い性能を示した.

今後の課題として, より幅広い問題クラスに対する理論解析と, ハイパーパラメータ最適化をはじめとした実問題への適用が挙げられる.

参考文献

- [Bergstra 11] Bergstra, J. S., Bardenet, R., Bengio, Y., and Kégl, B.: Algorithms for Hyper-Parameter Optimization, in *NIPS*, pp. 2546–2554 (2011)
- [Brochu 10] Brochu, E., Cora, V. M., and Freitas, de N.: A Tutorial on Bayesian Optimization of Expensive Cost Functions, with Application to Active User Modeling and Hierarchical Reinforcement Learning, Vol. abs/1012.2599, (2010)
- [Chiba 05] Chiba, K., Obayashi, S., Nakahashi, K., and Morino, H.: High-Fidelity Multidisciplinary Design Optimization of Wing Shape for Regional Jet Aircraft, in *Evolutionary Multi-Criterion Optimization*, pp. 621–635 (2005)
- [Golovin 17] Golovin, D., Solnik, B., Moitra, S., Kochanski, G., Karro, J., and Sculley, D.: Google Vizier: A Service for Black-Box Optimization, in *KDD*, pp. 1487–1495 (2017)
- [Goodfellow 14] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y.: Generative Adversarial Nets, in *NIPS*, pp. 2672–2680 (2014)
- [Hansen 03] Hansen, N., D Mller, S., and Koumoutsakos, P.: Reducing the Time Complexity of the Derandomized Evolution Strategy with Covariance Matrix Adaptation (CMA-ES), *Evolutionary computation*, Vol. 11, pp. 1–18 (2003)
- [Ilievski 17] Ilievski, I., Akhtar, T., Feng, J., and Shoemaker, C. A.: Efficient Hyperparameter Optimization for Deep Learning Algorithms Using Deterministic RBF Surrogates., in *AAAI*, pp. 822–829 (2017)
- [Lucic 18] Lucic, M., Kurach, K., Michalski, M., Gelly, S., and Bousquet, O.: Are GANs Created Equal? A Large-Scale Study, in *NIPS*, pp. 698–707 (2018)
- [Surjanovic 14] Surjanovic, S. and Bingham, D.: Virtual library of simulation experiments: Test functions and datasets., <https://www.sfu.ca/~ssurjano> (2014)
- [Yang 09] Yang, C., Kurahashi, S., Kurahashi, K., Ono, I., and Terano, T.: Agent-Based Simulation on Women’s Role in a Family Line on Civil Service Examination in Chinese History, *J. Artificial Societies and Social Simulation*, Vol. 12, (2009)

*1 <https://github.com/hyperopt/hyperopt>

表 1: 実験で用いるベンチマーク関数の名称, 定義式, 次元, 探索範囲. Shekel, Hartmann 関数に現れる係数は文献 [Surjanovic 14] から引用している.

関数名	定義式	次元	探索範囲
Sphere	$f(\mathbf{x}) = \sum_{i=1}^d x_i^2$	5	$[-5, 10]^5$
k -tablet ($k = d/4$)	$f(\mathbf{x}) = \sum_{i=1}^k x_i^2 + \sum_{i=k+1}^d (100x_i)^2$	5	$[-5, 10]^5$
RosenbrockChain	$f(\mathbf{x}) = \sum_{i=1}^{d-1} (100(x_{i+1} - x_i^2))^2 + (x_i - 1)^2$	5	$[-5, 10]^5$
Branin	$f(\mathbf{x}) = (x_2 - \frac{5.1}{4\pi^2}x_1^2 + \frac{5}{\pi}x_1 - 6)^2 + 10(1 - \frac{1}{8\pi})\cos(x_1) + 10$	2	$[-5, 10] \times [0, 15]$
Shekel ($m = 5$)	$f(\mathbf{x}) = -\sum_{i=1}^5 \left(\sum_{j=1}^4 (x_j - C_{ji})^2 + \beta_i \right)^{-1}$	4	$[0, 10]^4$
Hartmann	$f(\mathbf{x}) = -\sum_{i=1}^4 \alpha_i \exp\left(-\sum_{j=1}^6 A_{ij}(x_j - P_{ij})^2\right)$	6	$[0, 1]^6$

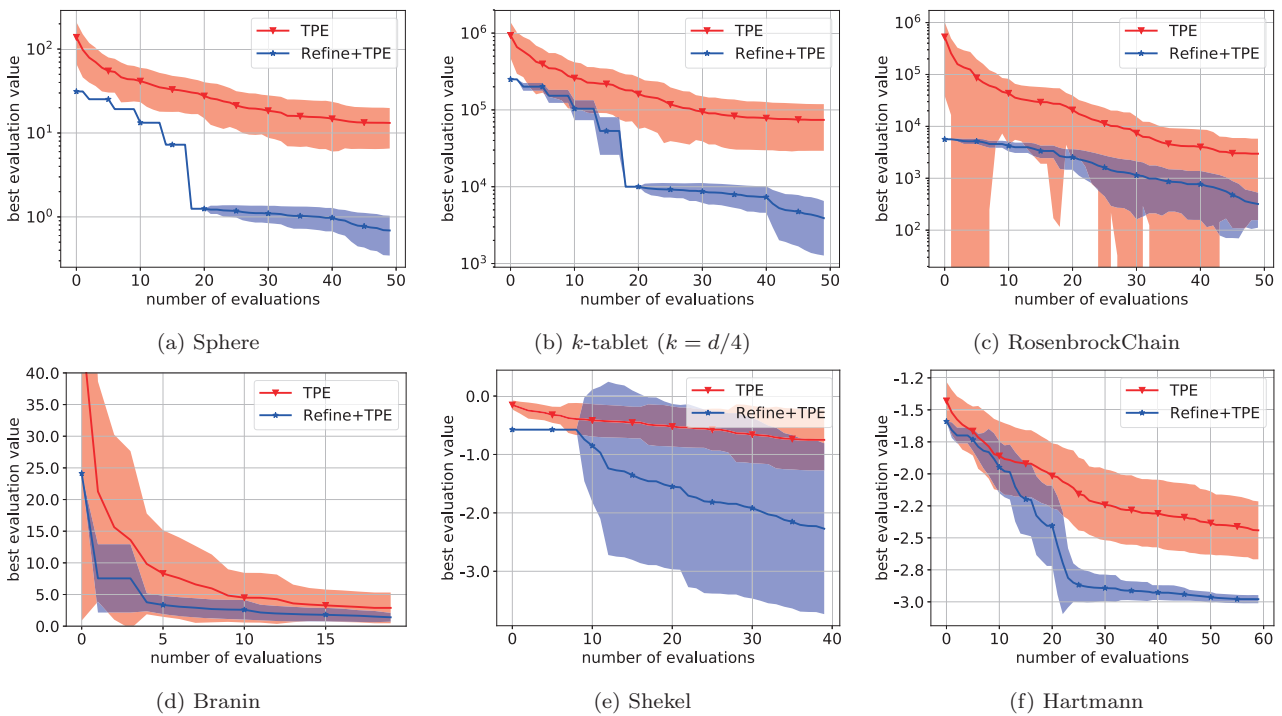


図 3: ベンチマーク問題に対する TPE と Refine+TPE の実験結果. 横軸は評価回数, 縦軸は最良評価値の 50 試行における平均と標準偏差を表している. 赤線は TPE, 青線は提案手法である Refine+TPE を表す.

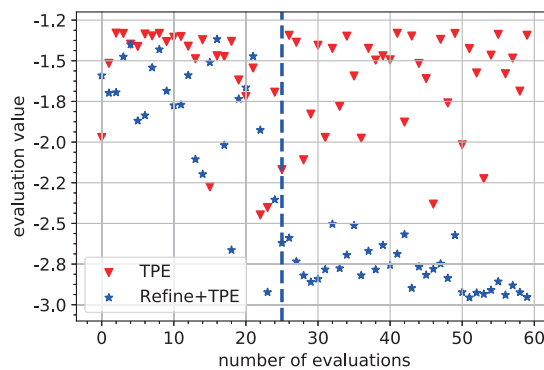


図 4: Hartmann 関数に対する TPE と Refine+TPE の典型的な挙動. 横軸は評価回数, 縦軸は評価値を表している. 赤点は TPE, 青点は提案手法である Refine+TPE を表す. また, 青の点線は探索空間の絞り込みが終了した時点を表す.