

## 特別研究報告

題目

多様な端末が混在する無線 LAN におけるベイズ最適化を用いた  
アクセスポイント制御方式

指導教員

若宮 直紀 教授

報告者

光安 海

2019 年 2 月 12 日

大阪大学 基礎工学部 情報科学科

平成 30 年度 特別研究報告

多様な端末が混在する無線 LAN におけるベイズ最適化を用いた  
アクセスポイント制御方式

光安 海

## 内容梗概

研究背景と目的. やったこと. その結果.

## 主な用語

無線 LAN, アクセスポイント制御, チャンネル割当, ベイズ最適化

## 目次

<b>1</b>	<b>はじめに</b>	<b>3</b>
<b>2</b>	<b>ベイズ最適化 [1]</b>	<b>4</b>
2.1	概要 . . . . .	4
2.2	獲得関数 . . . . .	4
2.2.1	Probability of Improvement (PI) . . . . .	4
2.2.2	Expected Improvement (EI) . . . . .	5
2.2.3	Gaussian Process Upper Confidence Bound (GP-UCB) . . . . .	5
<b>3</b>	<b>ベイズ最適化を用いたアクセスポイント制御方式</b>	<b>6</b>
3.1	入力の形式 . . . . .	6
3.2	チャネル割当とアクセスポイント割当のベイズ最適化 . . . . .	6
<b>4</b>	<b>シミュレーションによる評価</b>	<b>7</b>
4.1	簡易スループットモデル [2] . . . . .	7
4.2	シミュレーション設定 . . . . .	8
4.2.1	アクセスポイント選択アルゴリズム . . . . .	9
4.2.2	アソシエーション . . . . .	9
4.3	評価方法 . . . . .	9
4.4	シミュレーション結果 . . . . .	10
<b>5</b>	<b>おわりに</b>	<b>11</b>
	謝辞	12
	参考文献	13

## 1 はじめに

近年，無線 LAN の使い勝手の良さから，無線 LAN の普及が進んでいる．それに伴い，無線 LAN のアクセスポイントに接続する端末が多様化している．また，会社のオフィスや空港などでは，複数のアクセスポイントを複数の端末が共用する状況が多く見られる．その結果，複数のアクセスポイントで構成される無線 LAN 内に多様な端末が混在する環境が一般的になっている．

無線 LAN では，それぞれの端末が持っている情報を用いて自身の制御を行なっている．多様な端末が混在する環境では，無線 LAN の性能が低下する．例えば，端末はアクセスポイントの電波強度などの情報を用いて接続先アクセスポイントを選択するが，特定のアクセスポイントに接続が集中して，そのアクセスポイントに接続する端末のスループットが低下するなどの問題が起こりうる．

無線 LAN 全体を制御することで性能の向上が期待できる．その方法として，アクセスポイントと端末が協調して制御を行う，などが考えられる．アクセスポイントが端末を制御するには，接続する全ての端末に制御を行うためのプロトコルを実装する必要がある．しかしすでに多様な端末が存在することや今後新たに端末が現れることを考えると，全ての端末に同じようなプロトコルを実装するのは現実的に難しい．

多様な端末が混在する環境では，端末自身の制御の違いを吸収できるアクセスポイント制御方式が求められている．多様な端末が混在する環境において，無線 LAN 全体の振る舞いをアクセスポイントの制御によって変化するブラックボックスなシステムとみなす．ただし，端末のスループットに与える影響を抑えるために，試行回数を少なくする必要がある．ブラックボックスを少ない試行回数で最適値に近い値を得るためにベイズ最適化を用いる．この制御方式はアクセスポイントの情報と制御で完結するため，端末に特別な制御を実装する必要がなく，多様な端末が混在する環境において動作する．

本研究では，複数のアクセスポイントで構成される無線 LAN に多様な端末が混在する環境において，スループット向上のためのベイズ最適化を用いたアクセスポイント制御方式を提案し，その有効性を評価する．

## 2 ベイズ最適化 [1]

### 2.1 概要

ベイズ最適化は、目的関数の値を最大または最小にする変数の値を求める最適化アルゴリズムの一つである。ベイズ最適化では、目的関数の事前分布がガウス過程に従うと仮定し、値が得られているデータを基づき、ベイズの定理より事後分布を求める。また、探索 (不確実性が高い範囲での実験) と活用 (改善が見込める範囲での実験) の両方を考慮した獲得関数によって、次に実験する点を決定する。実験により新たに得られたデータを含めて事後分布を再び求め、実験を繰り返すことで、目的関数の値を最大または最小にする点に収束する。ベイズ最適化はグリッドサーチ法と比べて、効率良く計算できるため、計算コストが高い機械学習のハイパーパラメータ探索などに用いられる。

### 2.2 獲得関数

獲得関数は、ベイズ最適化において次に実験する点を決める役割を担っている。使用する獲得関数とそのパラメータによって探索と活用のバランスが変わる。以下にベイズ最適化で使用される獲得関数の例を3つ挙げる。

$i$  回目に実験した点を  $\mathbf{x}_i$ 、目的関数の値を  $f(\mathbf{x}_i)$  とする。ここでは、目的関数を最大化することを考え、 $t$  回の実験における最大値を  $f(\mathbf{x}^+)$  で表す。ただし、 $\mathbf{x}^+ = \operatorname{argmax}_{\mathbf{x}_i \in \mathbf{x}_{1:t}} f(\mathbf{x}_i)$  である。

#### 2.2.1 Probability of Improvement (PI)

$f(\mathbf{x})$  が  $f(\mathbf{x}^+)$  より改善される確率 PI は、探索と活用のトレードオフを制御するパラメータ  $\xi \geq 0$  を導入して式 (1) で求められる。

$$\text{PI}(\mathbf{x}) = \Phi\left(\frac{\mu(\mathbf{x}) - f(\mathbf{x}^+) - \xi}{\sigma(\mathbf{x})}\right) \quad (1)$$

$\Phi(\cdot)$  は正規累積分布関数、 $\mu(\cdot)$  と  $\sigma(\cdot)$  はそれぞれガウス過程から得られた平均と分散を表している。

PI では、パラメータ  $\xi$  によって greedy の度合いを緩和しているが、それでも局所的最適解に陥りやすい。

### 2.2.2 Expected Improvement (EI)

$f(\mathbf{x})$  が  $f(\mathbf{x}^+)$  より改善される期待値 EI は、探索と活用のトレードオフを制御するパラメータ  $\xi \geq 0$  導入して式 (2) で求められる.

$$\text{EI}(\mathbf{x}) = \begin{cases} (\mu(\mathbf{x}) - f(\mathbf{x}^+) - \xi)\Phi(Z) + \sigma(\mathbf{x})\phi(Z) & (\sigma(\mathbf{x}) > 0) \\ 0 & (\sigma(\mathbf{x}) = 0) \end{cases} \quad (2)$$

ただし,

$$Z = \begin{cases} \frac{\mu(\mathbf{x}) - f(\mathbf{x}^+) - \xi}{\sigma(\mathbf{x})} & (\sigma(\mathbf{x}) > 0) \\ 0 & (\sigma(\mathbf{x}) = 0) \end{cases}$$

$\phi(\cdot)$  と  $\Phi(\cdot)$  はそれぞれ標準正規分布の確率密度関数と累積分布関数を表している.  $\xi = 0.01$  だとうまくいきやすい [3].

### 2.2.3 Gaussian Process Upper Confidence Bound (GP-UCB)

ガウス過程で回帰した関数  $f(\mathbf{x})$  の信頼上限は、探索の重きを調整するパラメータ  $\nu$  を導入して式 (3) で求められる.

$$\text{GP-UCB}(\mathbf{x}) = \mu(\mathbf{x}) + \sqrt{\nu\tau_t}\sigma(\mathbf{x}) \quad (3)$$

ただし,

$$\tau_t = 2 \log \left( \frac{t^{\frac{d}{2}+2}\pi^2}{3\delta} \right)$$

$\delta$  は開区間  $(0,1)$  の任意の実数,  $d$  は  $\mathbf{x}$  の次元数である.  $1-\delta$  以上の確率で最適値に収束するが保証されている [4].

### 3 ベイズ最適化を用いたアクセスポイント制御方式

アクセスポイントが制御できる要素であるチャンネル割当とアクセスポイント割当を入力  $\mathbf{x}$  とし、無線 LAN の合計スループットを目的関数  $f(\mathbf{x})$  とする。ベイズ最適化を用いて、目的関数  $f(\mathbf{x})$  を最大にする入力  $\operatorname{argmax}_{\mathbf{x}} f(\mathbf{x})$  を求める。これにより無線 LAN の合計スループットを最大にするチャンネル割当とアクセスポイント割当を得る。

ベイズ最適化の計算は無線 LAN の全アクセスポイントに接続するコントローラで行う。コントローラがベイズ最適化を用いて得たチャンネル割当とアクセスポイント割当をアクセスポイントに送り、アクセスポイントが計算した無線 LAN の合計スループットをコントローラに送る。

#### 3.1 入力の形式

チャンネル割当とアクセスポイント割当を目的関数の入力として扱うために、アクセスポイントが利用できるチャンネルと端末が利用できるアクセスポイントにそれぞれラベルを付ける。このとき、目的関数への入力の次元数は (アクセスポイント台数 + 端末台数) となる。

#### 3.2 チャンネル割当とアクセスポイント割当のベイズ最適化

アクセスポイントのチャンネル割当を変える頻度を減らすために、チャンネル割当のベイズ最適化の中でアクセスポイント割当のベイズ最適化を行う。チャンネル割当のベイズ最適化とアクセスポイント割当のベイズ最適化をそれぞれ A,B とすると、フローチャートは図 1 である。あるチャンネル割当のもとでの最適なアクセスポイント割当をベイズ最適化を用いて求める。求めたアクセスポイント割当のもとでの最適なチャンネル割当をベイズ最適化を用いて決める。これらを繰り返し行い、最適なチャンネル割当とアクセスポイント割当を得る。

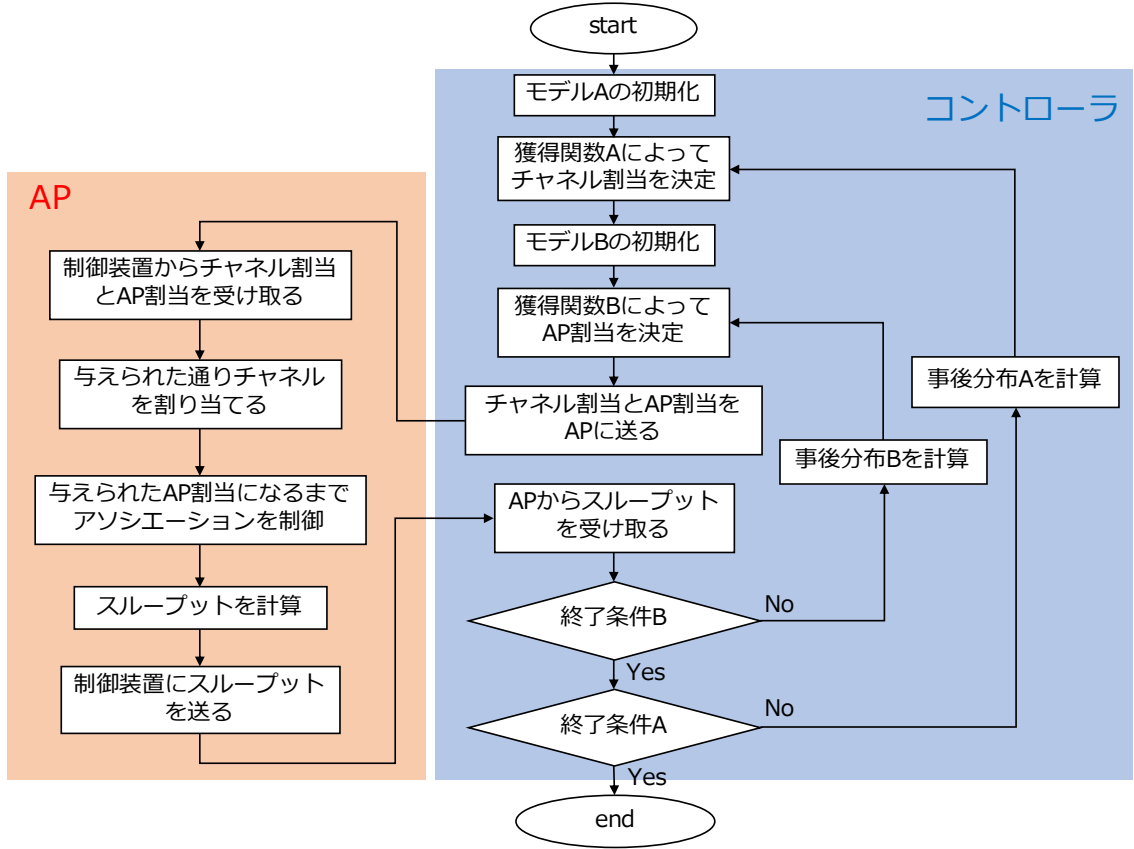


図 1: 提案手法のフローチャート

## 4 シミュレーションによる評価

整数の入力に対応するために文献 [5] の提案方法を採用したとき、ベイズ最適化のイテレーションに対するスループットの変化を調べる。

### 4.1 簡易スループットモデル [2]

ネットワークの端末  $i$  のスループット  $\eta_i$  は式 (4) で求められる。ただし、 $s$  は使用できるチャンネル数、 $m_i$  は端末  $i$  の隠れ端末の台数、 $m_i$  は端末  $i$  のさらし端末の台数である。

$$\eta_i = \frac{s}{m_i + 1} e^{-\frac{2n_i}{m_i + 1}} \quad (4)$$

端末数が  $n$  である無線 LAN の合計スループットは式 (5) で求められる。

$$S_n = \sum_{k=0}^n \eta_k \quad (5)$$



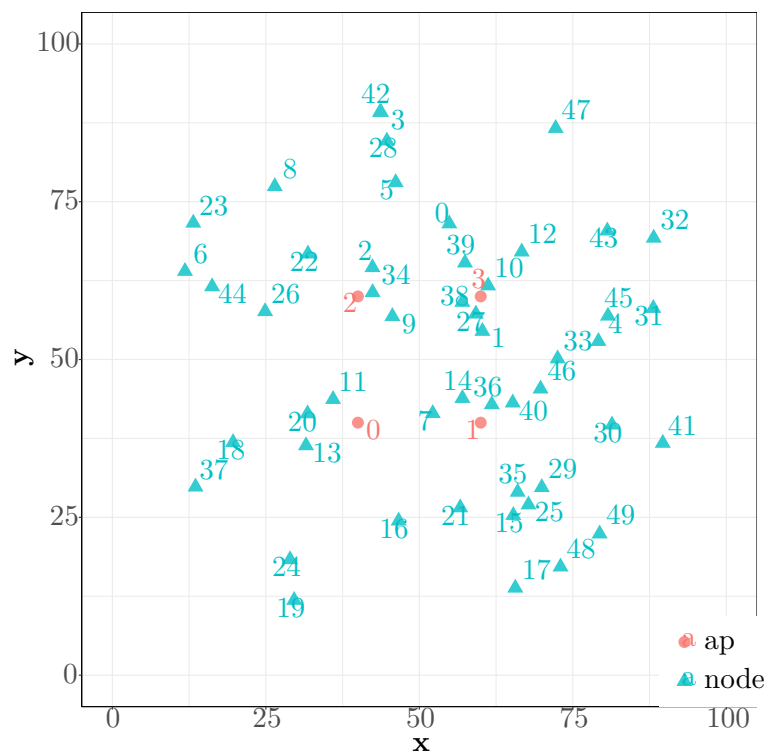


図 2: アクセスポイントと端末の配置図

## 4.2 シミュレーション設定

ベイズ最適化を実行する無線 LAN は図 2 である。アクセスポイントの位置は固定で，端末は少なくとも一台のアクセスポイントと接続できる位置にランダムに配置される。無線 LAN とベイズ最適化のパラメータはそれぞれ表 1 と表 2 である。

表 1: 無線 LAN のパラメータ

パラメータ	値
アクセスポイント台数	4 (台)
端末台数	50 (台)
チャンネル数	2
通信可能距離	30
干渉距離	45

表 2: ベイズ最適化のパラメータ

パラメータ	値
獲得関数	EI ( $\xi = 0.01$ )
カーネル	Matern 5/2
最大イテレーション	400 (回)

#### 4.2.1 アクセスポイント選択アルゴリズム

端末に適用するアクセスポイント選択アルゴリズムを以下に示す．アルゴリズム 1 は実際によく使われている．アルゴリズム 2 は一定以上のスループットを必要とする端末を想定している．今回の実験では閾値を 0.06 に設定している．

**アルゴリズム 1** 最も電波強度が高いアクセスポイントに接続

**アルゴリズム 2** スループットが閾値以下のとき接続先アクセスポイントを変更

#### 4.2.2 アソシエーション

簡易的なアソシエーションの動作は Alg.1 である．ただし，*max\_rejection* と *max\_request* はそれぞれアクセスポイントの最大アソシエーション拒否回数と端末の最大アソシエーション要求回数である．今回の実験では，*max\_rejection* は 5 回に固定し，*max\_request* は区間  $[1,10]$  の整数からランダムに決めている．

---

#### Algorithm 1 : アソシエーション

---

```

1: if max_rejection  $\geq$  max_request then
2:   ベイズ最適化の AP 割当に従う
3: else
4:   端末の AP 選択アルゴリズムに従う
5: end if

```

---

#### 4.3 評価方法

単純なチャネル割当アルゴリズムと比較し，提案手法の有効性を評価する．スループットの低下を抑えながら最適化を進めていることを強化するために，時間に対する平均スループットの変化を指標として考えている．

#### 4.4 シミュレーション結果

実験の結果は図 3, 4 である。初期値次第ではスループットが低い点でベイズ最適化が止まるため、工夫が必要だと考えている。また、良くないチャネル割当を入力したときにスループットが 1 付近まで落ちる。良くないチャネル割当では、AP 割当のベイズ最適化を進めても変化が少ないことから、チャネル割当に制限を設けることや早くにベイズ最適化を打ち切る工夫などが考えられる。

図 4 は 100 回の実験を行なったとき最終的に得られたスループットの最良値のヒストグラムである。図 4 の赤い線は参考値として、今までに行なった同じネットワークの配置での実験の中での最大値を示している。度数の高い階級がスループットの大きい方へ寄っている。初期値に対してスループットを向上できる確率が高いと言えるが、最適値にたどり着くことはできていない。また、スループットが低い階級ではベイズ最適化が途中で止まってしまうためにチャネル割当の探索が十分にできていなかった。

## 5 おわりに

やったこと．シミュレーション結果から得られた知見．今後の課題．

## 謝辞

本報告を終えるにあたり，日頃よりご指導，ご教授をいただきました大阪大学情報科学研究科の若宮直紀教授に心より感謝申し上げます．ならびに，終始丁寧なご指導をいただきました大阪大学情報科学研究科の橋本匡史助教に心より御礼申し上げます．さらに本報告を作成するにあたり，様々な助言をいただきました若宮研究室の方々に深く感謝申し上げます．

## 参考文献

- [1] E. Brochu, V. M. Cora, and N. De Freitas, “A Tutorial on Bayesian Optimization of Expensive Cost Functions, with Application to Active User Modeling and Hierarchical Reinforcement Learning,” Tech. Rep., 2010. [Online]. Available: <https://arxiv.org/pdf/1012.2599.pdf>
- [2] H. Wu, F. Zhu, Q. Zhang, Z. Niu, . A. Yongacoglu, and . L. Kleinrock, “IEEE 802.11a throughput performance with hidden nodes,” *IEICE Communications Express*, vol. 1, no. 3, pp. 101–106, 2000. [Online]. Available: <http://www.scalable-networks.com/>.
- [3] D. J. Lizotte, “Practical bayesian optimization,” Ph.D. dissertation, Edmonton, Alta., Canada, 2008, aAINR46365.
- [4] N. Srinivas, A. Krause, S. M. Kakade, and M. Seeger, “Gaussian Process Optimization in the Bandit Setting: No Regret and Experimental Design,” Tech. Rep. [Online]. Available: <https://arxiv.org/pdf/0912.3995.pdf>
- [5] E. C. Garrido-Merchán and D. Hernández-Lobato, “Dealing with Categorical and Integer-valued Variables in Bayesian Optimization with Gaussian Processes,” 2018.

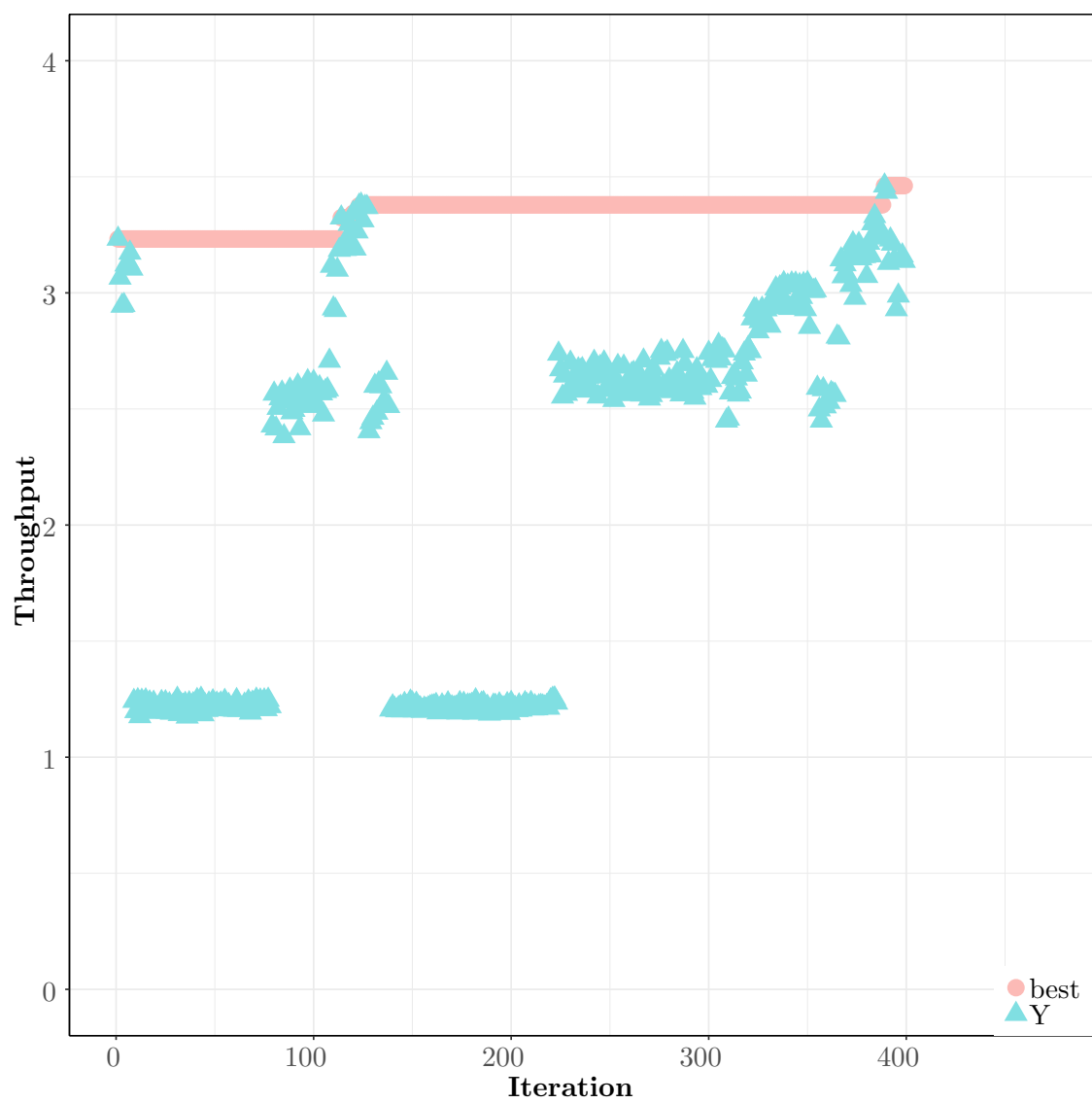


図 3: イテレーションに対するスループットの変化

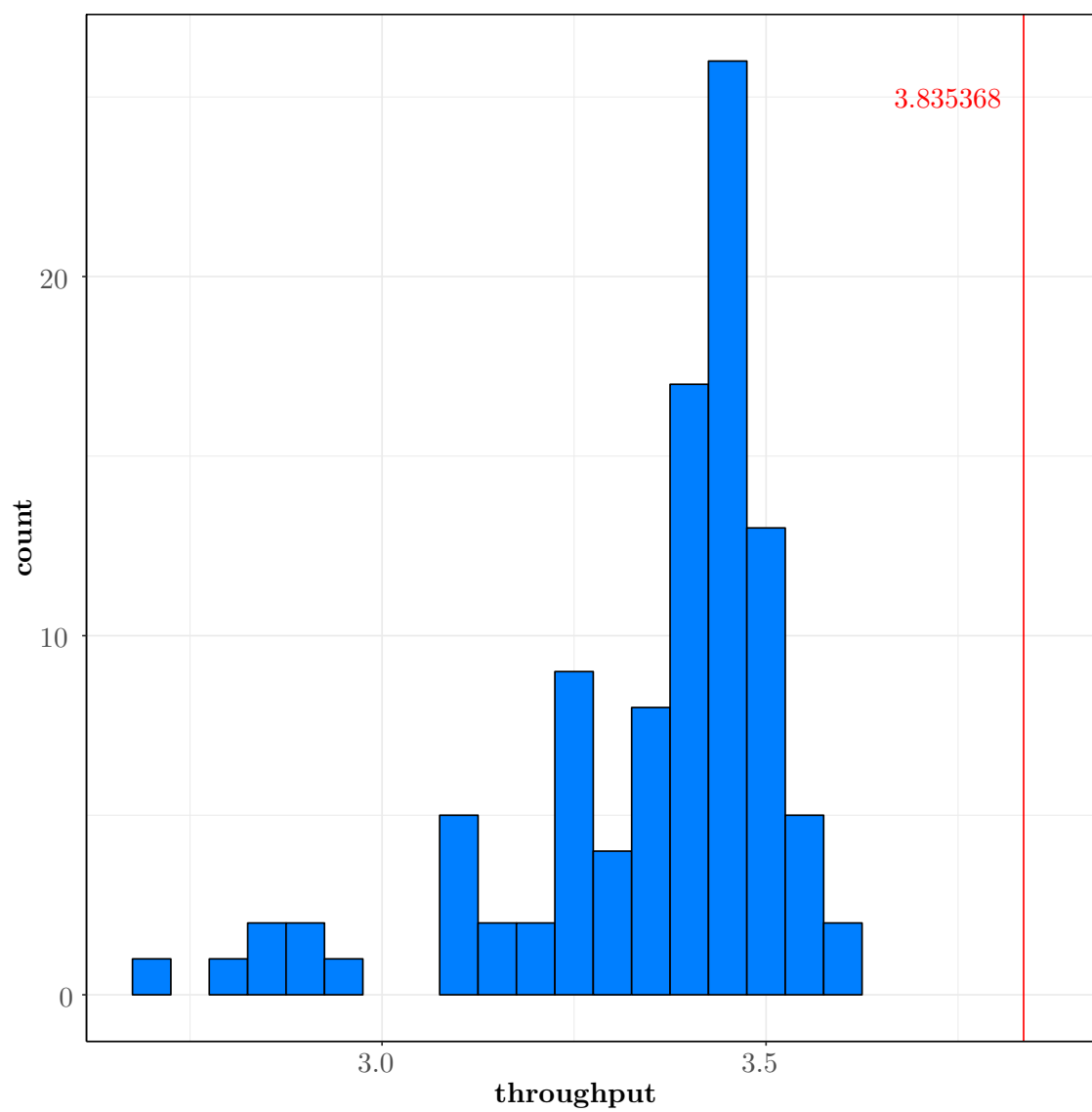


図 4: 100 回の実験における最良値のヒストグラム