The Sample Complexity of Multiple Change Point Identification under Bandit Feedback

Maximilian Graf †
Institut für Mathematik, Universität Potsdam, Potsdam, Germany
maximilian.graf@warwick.ac.uk

Victor Thuot †
INRAE, Mistea, Institut Agro, Univ Montpellier, Montpellier, France*
victor.thuot@ens-lyon.fr

† Equal contribution.

Abstract

We study multiple change point localization under bandit feedback. An unknown piecewise-constant function on a compact interval can be queried sequentially at adaptively chosen inputs, and each query returns a noisy evaluation of the function. The goal is to identify a prescribed number of discontinuities, known as change points, within a target precision η and confidence level $1-\delta$, while using as few samples as possible. We propose an adaptive algorithm that first detects intervals likely to contain change points and then refines their locations to precision η. We establish non-asymptotic upper bounds on its sample budget, together with corresponding lower bounds. Prior work shows that jump magnitudes alone determine the asymptotic sample complexity as $\delta \to 0$. We reveal that this picture is incomplete beyond this regime. We demonstrate, both empirically[2] and theoretically, that for general δ and η, the complexity is jointly governed by the jumps and the relative positions of the change points.

1 Introduction

Change point detection – identifying the thresholds at which a system's behavior changes abruptly – is a central problem in statistics and machine learning, with decades of theoretical development [Lai, 2001, Yu, 2020, Truong et al., 2020]. Many practical problems fall within the scope of change point detection. In materials science, one seeks the control parameters (e.g., temperature and pressure) at which a material undergoes a phase transition [Nikolaev et al., 2016, Sebastián et al., 2011]; in queuing and processing systems, one aims to detect abrupt changes in waiting times or throughput [Hung and Michailidis, 2007]; in machine learning, one studies the sensitivity of black-box models to hyperparameter changes [Lan et al., 2009, Hayashi et al., 2019]. In all these settings, evaluations are costly or time-consuming, and the input domain is continuous, so adaptive and sequential sampling strategies are particularly relevant. Moreover, reliable guarantees are required in practice: both the estimation precision and the probability of error must be controlled. This motivates a bandit formulation [Lattimore and Szepesvári, 2020], in which a learner queries the function at adaptively chosen inputs and aims to identify a prescribed number of change points, at a given precision η and confidence level $1-\delta$, while minimizing the total number of evaluations.

*Work done at INRAE, Mistea, Institut Agro, Univ Montpellier. Author now at UMPA, ENS Lyon.
[2]Code to reproduce all experiments is available at bandit_changepoint.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

In this manuscript, we consider the multiple change point detection problem in the bandit setting, as introduced in Hayashi et al. [2019] and recently studied in Lazzaro and Pike-Burke [2025a]. As in these works, we focus on the one-dimensional piecewise-constant model. Although many practical problems may require more involved settings and notions of change points, this model is the canonical theoretical baseline in the passive literature [Niu et al., 2016, Verzelen et al., 2023], and a natural starting point for the bandit setting, from which extensions to piecewise-linear, polynomial, or multivariate models can be built [Truong et al., 2020, Suk and Kpotufe, 2022, Manegueu et al., 2021]. To fix notation before discussing related work (a more detailed presentation is deferred to Section 2), we consider a piecewise-constant function $f : [0, 1] \to \mathbb{R}$ with m ($m \geq 1$) change points at positions $0 < x_1^* < \cdots < x_m^* < 1$, so that f is constant on $[x_{l-1}^*, x_l^*)$ for $l \in \{1, \ldots, m+1\}$, with $x_0^* = 0$ and $x_{m+1}^* = 1$. The learner sequentially and adaptively queries points $x_t \in [0, 1]$ and observes $y_t = f(x_t) + \varepsilon_t$, where ε_t is subGaussian noise; in particular, each new query may depend on past observations. Given a target number of change points $N \leq m$, a precision $\eta \in (0, 1)$, and an error rate $\delta \in (0, 1)$, the goal is to output N estimates within distance η of true change points with probability at least $1 - \delta$. The sample complexity, denoted by $\mathcal{T}$, is the total number of evaluations required to achieve this guarantee.

In Lazzaro and Pike-Burke [2025a], the authors propose `MCPI` (*Multiple Change Point Identification*), a method based on the stopping conditions from Garivier and Kaufmann [2016], and prove its asymptotic optimality in expectation. Their analysis is conducted in a discrete action space with K arms, which corresponds to a precision of $\eta \simeq 1/K$ in our continuous setting. They provide non-asymptotic upper bounds, which contain at least a linear[3] dependence on K, while the focus of their analysis lies on the asymptotic regime $\delta \to 0$. Indeed, `MCPI` identifies N change points with expected sample complexity satisfying $\lim_{\delta \to 0} \frac{\mathbb{E}[\mathcal{T}]}{\log(1/\delta)} = 8H^{(N)}_{\text{localize}}$, where

$$H^{(N)}_{\text{localize}} = \sum_{l=1}^{N} \frac{1}{\Delta^2_{(l)}} \ , \tag{1}$$

with $\Delta_i = f(x_i^*) - f(x_{i-1}^*)$ for $i \in \{1, \ldots, m\}$ denoting the jump at the i-th change point, and $|\Delta_{(1)}| \geq \cdots \geq |\Delta_{(m)}|$ denoting the decreasing rearrangement of jump magnitudes. Thus, the complexity is driven by the N largest jumps in the asymptotic regime $\delta \to 0$, while dependence on K disappears in the limit. The analysis of large-scale problems (K large) remains open.

For comparison, consider the batch change point detection problem, in which the learner evaluates f once at each point of a fixed grid [Lai, 2001, Yu, 2020]. In that setting, the relevant difficulty measure is the energy of each change point, defined as $\mathcal{E}_i^2 := s_i \Delta_i^2$, where s_i is the minimal spacing to neighboring change points (see Equation (4) in Section 2). One of our contributions is to show that energy also matters in the active setting. For instance, the guarantees in Verzelen et al. [2023] imply a uniform-sampling detection threshold, and hence a sample-complexity term of order

$$H^{(m)}_{\text{detect}} = \max_{i=1,\ldots,m} \frac{1}{\mathcal{E}_i^2}, \tag{2}$$

which is always at least as large as $\frac{1}{2} H^{(N)}_{\text{localize}}$; it can in fact be much larger. For instance, consider change points with comparable jump size Δ and local spacing s (with $s \leq 1/m$). Then $H^{(m)}_{\text{detect}} \asymp \frac{1}{s\Delta^2}$, whereas $H^{(N)}_{\text{localize}} \asymp \frac{N}{\Delta^2}$. When s is small, detection may dominate the budget, showing that the rate (1) can miss an important non-asymptotic effect. This suggests a richer interplay between jump magnitudes and local spacings than what is captured by the asymptotic jump-only rate in (1).

We consider a continuous action space that avoids the restrictive assumption $\eta < \min_{i=1}^{m} s_i$ required by discrete methods such as Lazzaro and Pike-Burke [2025a]. In the discrete setting, this induces a linear dependence on $1/\eta$, which can become prohibitive when high localization precision is required. Our algorithm adapts to local spacings s_i, which is practically significant when change points may be arbitrarily close and the spacing is unknown. By decomposing detection and localization, we show that the dependence on precision η is logarithmic, namely $\log(1/\eta)$, which is exponentially better than the linear dependence on the discretization grid $K \simeq 1/\eta$ inherent to discrete approaches. Deriving guarantees that explicitly capture the optimal dependence on both η and δ in all regimes is an open question and a key motivation for our work.

[3] Additionally, a polynomial dependence on K (namely, K^4), independent of δ, appears to be hidden in the non-explicit terms $T_1'(v)$ of their Prop. 5.6.

We illustrate the energy-based complexity in **Experiment 1** (Figure 1) with two change points ($\Delta_1 = -\Delta_2 = 1$), standard noise, and varying spacing s. We compare our Algorithm 2 (`LCP`) to `MCPI` from Lazzaro and Pike-Burke [2025a].[4] The results validate the theory: as s increases, sample complexity of `LCP` decreases roughly as $1/s$, matching the prediction $H_{\text{detect}} \asymp 1/(s\Delta^2)$. In contrast, `MCPI` is insensitive to s because its rate is dominated by a constant term proportional to $1/\eta$. Then, adaptation to local spacings is essential, as it allows us to outperform `MCPI` for moderate values of s.

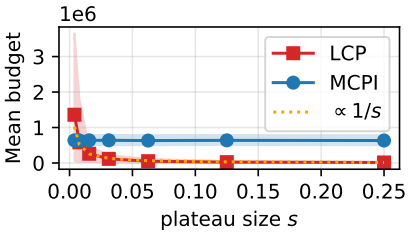


Figure 1: **Experiment 1**. We consider two change points. We set $x_1^* \sim \mathcal{U}(0, 1/2)$ for each Monte Carlo run and $x_2^* = x_1^* + s$ with s varying and 1000 Monte Carlo runs. We chose $\Delta_1 = -\Delta_2 = 1$ and compare `LCP` (Algorithm 2) with parameters $\eta = 2^{-11}$, $\delta = 0.05$ and $\delta_{\text{explore}} = 1$ to algorithm `MCPI`.

Open questions. These observations raise two fundamental questions on the sample complexity of multiple change point identification: How does it depend on jump magnitudes and local spacings, i.e., when is it governed by the energies $\mathcal{E}_i$ rather than only by the jumps Δ_i? And what is the optimal dependence on the precision η and the confidence parameter δ, in particular for high precision (η small)?

Contributions. In this manuscript, we answer these questions.

1. We propose `LocalizeChangePoints` (Algorithm 2), a new algorithm for multiple change point identification under bandit feedback. It decomposes the task into (i) detecting intervals likely to contain change points and (ii) sequential refinement by binary search. We prove non-asymptotic sample-complexity guarantees that depend on both jump magnitudes and local spacings through $\mathcal{E}_i$ and Δ_i. In particular, Theorem 1 states that the budget $\mathcal{T}$ for localizing N change points among m is bounded, up to logarithmic terms independent of δ and η,[5] as
$$\mathbb{E}[\mathcal{T}] \underset{\log}{\leq} H^{(m)}_{\text{detect}} + H^{(N)}_{\text{localize}} \log\left(\frac{1}{\delta\eta}\right), \qquad \text{see (1), (2)}$$
$$\mathcal{T} \underset{\log}{\leq} H^{(m)}_{\text{detect}} \log\left(\frac{1}{\delta}\right) + H^{(N)}_{\text{localize}} \log\left(\frac{1}{\delta\eta}\right) \qquad \text{w.h.p. } 1-\delta.$$
2. Additionally, we derive information-theoretic lower bounds (Theorem 2). On a local-minimax class of instances, we prove that the bounds above are rate-optimal in all parameters, including δ and η. This shows that the asymptotic rate in (1) is overly optimistic.
3. Finally, we validate our results numerically on synthetic data.

Batch change point detection. Change point detection is one of the classical problems in statistics, with a rich literature spanning cumulative-sum (CUSUM) procedures [Hinkley, 1970], binary segmentation [Scott and Knott, 1974], wild binary segmentation (WBS) [Fryzlewicz, 2014], and penalized least-squares methods [Wang et al., 2020, Verzelen et al., 2023]; see Lai [2001], Yu [2020], Truong et al. [2020] for comprehensive reviews. Our algorithm is directly inspired by two classical ideas from this literature: multiscale testing as in the foundation work on WBS [Fryzlewicz, 2014], to identify candidate intervals likely to contain a change point [Kovács et al., 2023, Pilliat et al., 2023], and binary search, to localize it to precision η. It is also well established that the relevant difficulty measure is the energy $\mathcal{E}_i^2 = s_i\Delta_i^2$ of each change point [Yu, 2020, Verzelen et al., 2023], an observation that we extend to the bandit setting. The problem has also been studied in multivariate [Wang and Samworth, 2018] and kernel-based [Arlot et al., 2019] settings. The piecewise-constant model has proved a fruitful baseline for broader change point methods [Niu et al., 2016].

A complementary family of methods detects change points by minimizing a penalized cost over all possible segmentations, with BIC used to select their number [Zou et al., 2014, Niu et al., 2016]. PELT [Killick et al., 2012] solves this problem exactly in $O(n)$ time (under mild conditions)

[4] In Appendix F we show how to transform our setting into a discrete problem, for comparison with the `MCPI` Algorithm.

[5] Typically $\underset{\log}{\leq}$ hides here at most a $\log(H^{(m)}_{\text{detect}})$ multiplicative factor.

via pruned dynamic programming, and Zou et al. [2014] extend the penalized formulation to a nonparametric, distribution-free setting. Because these methods process the full data sequence, their main contribution is computational efficiency rather than sample complexity. Whether such ideas – or related clustering-based approaches [Banerjee et al., 2026] – can be adapted to the active-sampling setting is an interesting open question.

Change point detection in piecewise-stationary bandits. A related but distinct stream of literature studies piecewise-stationary bandits [Garivier and Moulines, 2011], where the reward distributions of a multi-armed bandit change at unknown time steps. Two main tasks are considered: quickest detection, which aims to raise an alarm as soon as possible after a change occurs [Veeravalli et al., 2024, Zhang and Mei, 2023, Xu and Mei, 2023], and regret minimization, where the goal is to adapt to the new environment in order to minimize cumulative regret [Cao et al., 2019, Besson et al., 2022, Mukherjee, 2022]. This more established literature illustrates that the piecewise-constant assumption is a natural starting point: Suk and Kpotufe [2022] go beyond it by targeting only *significant* arm switches, and Manegueu et al. [2021] extend the framework further to piecewise-linear and piecewise-polynomial reward regimes. Our setting differs from this line of work in two fundamental respects: our model is stationary, and the change points are *spatial* features of a fixed function f rather than temporal. Our model is therefore a specific instance of the well-established continuous bandit framework – see, e.g., unimodal bandits [Combes and Proutiere, 2014], continuous dueling bandits [Kumagai, 2017], and pure-exploration problems over continuous domains [Bubeck et al., 2011].

Active change point detection. Active and sequential change point detection studies settings in which the sampling locations are chosen adaptively rather than fixed in advance. Early foundations for this perspective appear in adaptive change point estimation [Hall and Molchanov, 2003, Lan et al., 2009], where multistage sampling is used to improve localization rates. In the bandit setting, Hayashi et al. [2019] formulates active change point detection explicitly, and recent fixed-budget and fixed-confidence analyses for piecewise-constant bandits [Lazzaro and Pike-Burke, 2025b,a] provide instance-dependent guarantees based on jump magnitudes. In particular, Lazzaro and Pike-Burke [2025a] combines this formulation with Track-and-Stop ideas from Garivier and Kaufmann [2016] and proves asymptotic rates proportional to $\sum_{l=1}^{N} 1/\Delta_{(l)}^2$. Our work complements this line by focusing on non-asymptotic guarantees and by highlighting the additional role of spacing between Change Points. We highlight the strong connection to other structured bandit problems, such as monotone bandits [Cheshire et al., 2020] or clustering with bandit [Yang et al., 2024, Graf et al., 2025].

Outline. The rest of the paper is organized as follows. In Section 2, we introduce the formal setting and notation. In Section 3, we present our algorithm and its guarantees. In Section 4, we derive information-theoretic lower bounds. In Section 5, we illustrate our results numerically on synthetic data. Finally, in Section 6, we discuss the implications of our results, possible extensions, and limitations. All proofs are deferred to the appendix.

2 Problem formulation and notation

A summary of the notation is given in Table 1 (Appendix A). We consider an unknown piecewise-constant function $f : [0, 1] \to \mathbb{R}$ with $m \geq 1$ change points at positions $0 < x_1^* < \cdots < x_m^* < 1$. The change points are the discontinuities of f, so f is constant on each interval $[x_{l-1}^*, x_l^*)$ for $l \in \{1, \ldots, m+1\}$, with the conventions $x_0^* = 0$ and $x_{m+1}^* = 1$. Both f and the true number of change points m are unknown to the learner.

We parametrize f as

$$f(x) = \mu_0 + \sum_{i=1}^{m} \Delta_i \mathbb{1}_{x \geq x_i^*} \,, \tag{3}$$

where $\mu_0 \in \mathbb{R}$ is the baseline level and $\Delta_i = f(x_i^*) - f(x_{i-1}^*)$ for $i \in \{1, \ldots, m\}$ is the jump at the i-th change point. For any compact interval $I = [l(I), r(I)]$, we introduce the notation $\Delta(I) = f(r(I)) - f(l(I))$ for the jump magnitude across the interval I. We assume $\Delta_i \neq 0$ for all i, so every change point is identifiable. Let $|\Delta_{(1)}| \geq \cdots \geq |\Delta_{(m)}|$ denote the decreasing ordering of jump magnitudes. These jumps govern asymptotic sample-complexity rates in Lazzaro and Pike-Burke [2025a], through the complexity $H_{\text{localize}}^{(N)} = \sum_{i=1}^{N} 1/\Delta_{(i)}^2$ (Equation (1)). Without

loss of generality (by rescaling with a known bound), we assume $|\Delta_i| \leq 1$ for all i, which avoids unnecessary constants.

Spacing between change points also plays a central role. For $i \in \{1, \dots, m-1\}$, let $\vartheta_i = x^*_{i+1} - x^*_i$ denote the spacing after the i-th change point. By convention, define $\vartheta_0 = \vartheta_m = 1$. We then define the local spacing by $s_i = \vartheta_{i-1} \wedge \vartheta_i$ for $i = 1, \dots, m$, and the associated energies by

$$\mathcal{E}_i^2 = s_i \Delta_i^2 \qquad \text{for } i = 1, \dots, m. \tag{4}$$

The quantity s_i plays the role of sparsity level, and the quantity $\mathcal{E}_i$ captures the effect of local spacing on the difficulty of detecting change point i, in line with the batch literature [Yu, 2020]. The interplay between these quantities is captured in the complexity term $H^{(m)}_{\mathrm{detect}} = \max_{i=1,\dots,m} 1/\mathcal{E}_i^2$ (Equation (2)). Observe that for $m = N = 1$, one has $H_{\mathrm{detect}} = H_{\mathrm{localize}}$.

We work in a bandit setting: at each round $t \geq 1$, the learner chooses $x_t \in [0, 1]$ and observes $y_t = f(x_t) + \varepsilon_t$, where $(\varepsilon_t)_{t \geq 1}$ are i.i.d. 1-subGaussian random variables, i.e., $\mathbb{E}[\varepsilon_t] = 0$ and $\mathbb{E}[\exp(\lambda \varepsilon_t)] \leq \exp(\lambda^2/2)$ for all $\lambda \in \mathbb{R}$. This includes, for instance, Gaussian and bounded noise.[6] The pair (f, ε) defines an environment, denoted by ν.

Sampling is sequential and adaptive: each query x_t may depend on past observations. An algorithm π consists of three components: (i) a sampling rule for choosing next query, (ii) a stopping rule, and (iii) a decision rule that outputs estimated change points. These components may be randomized. The (random) sample complexity of π, denoted by $\mathcal{T}_\pi$, is the total number of evaluations of f. We write $\mathbb{P}_{\pi,\nu}$ and $\mathbb{E}_{\pi,\nu}$ for probability and expectation under environment ν and algorithm π.

The learner is given a precision $\eta \in (0, 1)$, a confidence parameter $\delta \in (0, 1)$, and a target number of change points $N \in \{1, \dots, m\}$. The objective is to identify N distinct change points within distance η of true change points with probability at least $1 - \delta$, while minimizing the sample complexity. Note that the learner can select any of the m change points, as long as it localizes N distinct change points.

Definition 1 ((δ, η, N)-correctness)**.** *An algorithm π is (δ, η, N)-correct if, for every environment ν with at least N change points,*

$$\mathbb{P}_{\pi,\nu}\left(\exists 1 \leq i_1 < \cdots < i_N \leq m : \ \forall l \in [N], |\hat{x}_l - x^*_{i_l}| \leq \eta\right) \geq 1 - \delta \ , \tag{5}$$

where $\hat{x}_1, \dots, \hat{x}_N$ are the estimates returned by π.

3 Algorithmic method and guarantees

3.1 Detection of change points

We describe the detection subroutine at the core of our method. Given a confidence level δ and a budget T, `DetectIntervals` (Algorithm 1) returns a set $\mathcal{I}$ of intervals with disjoint interiors. Using multiscale endpoint tests, the procedure guarantees that every returned interval contains at least one change point, with failure probability at most δ (Lemma 1).

Procedure. For each depth $d \in \{1, \dots, d_{\max}\}$ (with $d_{\max}$ defined in Line 1), the domain is partitioned into $n_d = 2^d$ dyadic intervals of equal length. Each endpoint is sampled T_d times (Line 3). Interval $[i/2^d, (i+1)/2^d]$ is flagged when the empirical jump between its endpoints exceeds the Hoeffding threshold β_d (Line 8), which accounts for the multiple testing across depths, intervals, and endpoints.[7] Flagged intervals are aggregated across depths; if a newly flagged interval is strictly contained in an existing one, the larger one is removed (Line 10). This pruning keeps $\mathcal{I}$ disjoint while favoring the finest detected intervals.

Lemma 1. *Let $0 < \delta < 1/4$. On an event ξ with $\mathbb{P}(\xi) \geq 1 - \delta$, Algorithm 1 only outputs intervals that contain at least one change point. In addition, there exists a sufficiently large numerical constant $c > 0$ such that for any change point x^*_i (with $i \in \{1, \dots, m\}$), if $T \geq c \cdot \omega^a_i(\delta) \cdot \frac{1}{\mathcal{E}_i^2}$, with $\omega^a_i(\delta) = (\log(\mathcal{E}_i^{-2}) \vee 1) \cdot \log\left(\frac{\log(\mathcal{E}_i^{-2}) \vee e}{s_i \cdot \delta}\right)$, then on the same event ξ, there exists an interval $I \in \mathcal{I}$ such that $l(I) < x^*_i \leq r(I)$ (the change point may coincide with the right endpoint of I) and $|I| \leq \frac{s_i}{2}$. In particular, if $T \geq \max_{i=1,\dots,m} c \cdot \omega^a_i(\delta) \cdot \frac{1}{\mathcal{E}_i^2}$, then on the event ξ, for any change point $i \in [m]$, there exists a unique interval $I \in \mathcal{I}$ containing x^*_i, and this interval has a length smaller than $\frac{s_i}{2}$.*

[6] The analysis extends to σ-subGaussian noise by a standard rescaling.

[7] If $T_d = 0$, β_d is set to $+\infty$ by convention.

Algorithm 1 `DetectIntervals`

Require: δ confidence parameter, T budget
Ensure: $\mathcal{I}$ set of disjoint intervals
1: $\mathcal{I} \leftarrow \emptyset$, $d_{\max} \leftarrow \lfloor \log_2(T/\log(1/\delta)) \rfloor$
2: **for** $d = 1, \dots, d_{\max}$ **do** ▷ Multiple-scale loop
3: $n_d \leftarrow 2^d$, $T_d \leftarrow \left\lfloor \frac{T}{d_{\max}(n_d+1)} \right\rfloor$, $\beta_d \leftarrow \sqrt{\frac{8}{T_d} \log\left(\frac{2d_{\max}(n_d+1)}{\delta}\right)}$
4: **for** $i = 0, 1, \dots, n_d$ **do**
5: sample T_d times from $x_i^{(d)} = i/n_d$, store means as $\hat{y}_i^{(d)}$ ▷ Sample uniformly at endpoints
6: **end for**
7: **for** $i = 1, \dots, n_d$ **do**
8: **if** $|\hat{y}_i^{(d)} - \hat{y}_{i-1}^{(d)}| > \beta_d$ **then** ▷ Test for presence of jump between adjacent endpoints
9: remove from $\mathcal{I}$ any set containing $[x_{i-1}^{(d)}, x_i^{(d)}]$ ▷ Pruning step
10: add $[x_{i-1}^{(d)}, x_i^{(d)}]$ to $\mathcal{I}$, ▷ Update $\mathcal{I}$
11: **end if**
12: **end for**
13: **end for**

Guarantees and energy scaling. For intuition on Lemma 1, consider for change point i its *natural depth* $d_i^\star \asymp \log_2(1/s_i)$: at this depth, one dyadic interval isolates x_i^* and has left-right jump $|\Delta_i|$. Detection requires $|\Delta_i| \gtrsim \beta_{d_i^\star}$, i.e., $T_{d_i^\star} \gtrsim \Delta_i^{-2} \log(1/\delta)$. Since $T_{d_i^\star} \asymp s_i T$ up to logarithmic factors, this yields $T \gtrsim \mathcal{E}_i^{-2} \log(1/\delta)$, which explains the $H^{(m)}_{\text{detect}}$ term in the upper bound.

3.2 Main algorithm and guarantees

Algorithm 2 takes as input the target number of change points N, the confidence level δ, and the precision η. Importantly, `LCP` does not require knowledge of m; only the target $N \leq m$ is needed as input. In addition, it uses a tuning parameter δ_{explore}, which controls the probability of failure of the exploration phase and, therefore, the budget. No other problem-dependent parameter is required.

The algorithm follows a doubling schedule. At stage k (initialized in Line 1), it is given budget $T_k = 2^{k+2}$ and runs four steps that mimic the oracle strategy described above. If, at the end of the stage, all N candidates are certified with error probability at most $\delta^{(k)}$, the algorithm returns the estimates $\mathcal{C} = \{c_1^{(k)}, \dots, c_N^{(k)}\}$ (Line 13). Otherwise, it moves to stage $k+1$, doubles the budget, and repeats (Line 17). Before stating the main guarantees of Algorithm 2, we describe each step.

At a generic stage with budget T_k, the four steps are the following.

(i) Detection. In Line 3, `DetectIntervals` (Algorithm 1) constructs a set $\mathcal{I}$ of disjoint intervals, each certified to contain at least one change point. This step is run with confidence level $\delta_{\text{explore}}/4$ and is described in Section 3.1.

(ii) Jump estimation. In Line 5, `EstimateJumps` (Algorithm 3) estimates the jump magnitudes on the candidate intervals and keeps the N most informative ones. The procedure is designed so that, with high probability, these estimates are accurate up to a constant multiplicative factor. It returns N intervals $\mathcal{J}^{(k)} = \{J_1^{(k)}, \dots, J_N^{(k)}\}$ together with the corresponding jump estimates $\mathcal{G}^{(k)} = \{\hat{\Delta}_1^{(k)}, \dots, \hat{\Delta}_N^{(k)}\}$, where $\hat{\Delta}_v^{(k)}$ estimates $|\Delta(J_v^{(k)})|$. The intervals are ordered by decreasing estimated jump, so that $\hat{\Delta}_1^{(k)} \geq \hat{\Delta}_2^{(k)} \geq \cdots \geq \hat{\Delta}_N^{(k)}$. `EstimateJumps` is a standard adaptive estimation routine. We defer its description to Appendix D.2; its guarantees are stated in Lemma 3.

(iii) Refinement of localization. We use `SHB` (*Sequential Halving with Backtracking*) from Lazzaro and Pike-Burke [2025b], which localizes a single change point up to precision η under a fixed budget constraint. The method consists of a binary-search procedure, with a backtracking mechanism as in Cheshire et al. [2020]. In particular, when applied to an interval containing a single change point of jump magnitude Δ, a budget of order $\Delta^{-2} \log(1/(\delta\eta))$ suffices to return an estimate within distance

η of the true change point with probability at least $1-\delta$ [Lazzaro and Pike-Burke, 2025b]. For completeness, we restate these guarantees in Lemma 4, and the routine itself in Algorithm 4.

In Line 9, we apply SHB independently to each selected interval. The budget is allocated proportionally to the inverse squared estimated jump, so interval $J_v^{(k)}$ receives budget $T_v^{(k)}$ with

$$T_v^{(k)} = \left\lfloor \alpha_v^{(k)} \cdot 2^k \right\rfloor \vee 1 \text{ , with } \qquad \alpha_v^{(k)} = \frac{\left(\hat{\Delta}_v^{(k)}\right)^{-2}}{\sum_{v'=1}^{N} \left(\hat{\Delta}_{v'}^{(k)}\right)^{-2}} \ . \tag{6}$$

(iv) Verification. In Line 10, VerifyCP (Algorithm 5) tests whether the candidates $(c_v^{(k)})_{v\in[N]}$ lie within distance η of N distinct change points. For each v, it performs a two-sample test at the points

$$l_v^{(k)} := \max\left(\min J_v^{(k)}, c_v^{(k)} - \eta\right), \qquad r_v^{(k)} := \min\left(\max J_v^{(k)}, c_v^{(k)} + \eta\right), \tag{7}$$

using confidence level $\delta^{(k)} = \frac{3\delta}{2\pi^2 N k^2}$. It draws $T_v^{(k)}/2$ samples at each endpoint and compares the empirical jump to the threshold $\sqrt{\frac{16}{T_v^{(k)}} \log\left(\frac{2}{\delta^{(k)}}\right)}$. If the test is positive, then $c_v^{(k)}$ is certified to be within distance η of a change point with probability at least $1-\delta^{(k)}$. The routine is given in Algorithm 5, and its guarantees are stated in Lemma 5.

Remark 1. *The verification step (iv) can not be performed with the jump estimation from Step (ii). Indeed, an interval returned in Step (i) may contain many change points with small jumps, whose cumulative effect creates a large jump between the left and right endpoints. In such cases, the refinement step is not guaranteed to converge to a change point, and verification is necessary.*

Theorem 1. *Let $N \le m$, $0 < \delta < 1/4$, $0 < \eta < 1/4$, and $\delta_{\text{explore}} \le 1/4$. Consider any environment ν with $m \ge N$ change points, and run Algorithm 2 with input parameters $(N, \delta, \eta, \delta_{\text{explore}})$.*

1. *(Correctness) Algorithm 2 is (δ, η, N)-correct (see (5)). With probability at least $1-\delta$, Algorithm 2 returns N points $0 \le c_1 < c_2 < \cdots < c_N \le 1$ such that $|c_i - x_{l_i}^*| \le \eta$ for $i = 1, \ldots, N$, and $1 \le l_1 < l_2 < \cdots < l_N \le m$.*
2. *(High-probability bound) If we set $\delta_{\text{explore}} = \delta$, then there exists an event ξ of probability at least $1-\delta$ on which the output is correct (Definition 1), and the budget $\mathcal{T}$ satisfies*

$$\mathcal{T} \le c \cdot \left(\omega_1 H_{\text{detect}}^{(m)} \left(\log\left(\frac{1}{\delta}\right) + \omega_2\right) + H_{\text{localize}}^{(N)} \left(\log\left(\frac{1}{\delta\eta}\right) + \omega_3\right)\right).$$

3. *(Expectation bound) If we set $\delta_{\text{explore}} = 1/4$, then the expected budget $\mathbb{E}[\mathcal{T}]$ is bounded by*

$$\mathbb{E}[\mathcal{T}] \le c \cdot \left(\omega_1 H_{\text{detect}}^{(m)} \omega_2 + H_{\text{localize}}^{(N)} \left(\log\left(\frac{1}{\delta\eta}\right) + \omega_3\right)\right).$$

In these bounds, $c > 0$ is a numerical constant, and $\omega_1, \omega_2, \omega_3$ are logarithmic terms defined as $\omega_1 = \log(H_{\text{detect}}^{(m)})$, $\omega_2 = \max_{i=1,\ldots,m}\left\{\log\left(\log\left(\mathcal{E}_i^{-2}\right) \vee e\right) + \log\left(1/s_i\right)\right\}$, and $\omega_3 = \log\left(N \cdot \left(\log\left(H_{\text{localize}}^{(N)}\right) \vee 1\right)\right)$. See (2), (1) for definitions of $H_{\text{detect}}^{(m)}$ and $H_{\text{localize}}^{(N)}$.

Intuition. To interpret $H_{\text{localize}}^{(N)} = \sum_{i=1}^{N} \Delta_{(i)}^{-2}$, consider an oracle that knows the jumps $(\Delta_i)_{i=1}^m$ and provides N disjoint intervals each containing one target change point. Running N parallel binary searches then costs $H_{\text{localize}}^{(N)} \log(1/(\delta\eta))$ – the pure localization cost. The main challenge is that these intervals are unknown; acquiring them adaptively requires detecting that a region contains a change point. Our main message is that it comes at an additional cost of $\mathcal{E}_i^{-2} \log(1/\delta)$ per change point. The total complexity thus splits into a detection cost $H_{\text{detect}}^{(m)}$ and a localization cost $H_{\text{localize}}^{(N)} \log(1/(\delta\eta))$. Complete proofs are given in Section D.

Sketch of proof. By step (iv), each stage-k acceptance via VerifyCP fails with probability at most $\delta^{(k)}$ (Lemma 5), so the union bound $\sum_k N\delta^{(k)} \le \delta$ implies (δ, η, N)-correctness.
We now sketch the budget bound by following the four-step procedure at stage k.

Algorithm 2 `LocalizeChangePoints`

Require: N number of change points, δ confidence parameter for correctness, η precision parameter, δ_{explore} confidence parameter for budget control
Ensure: $\mathcal{C}$ estimated change points

$k \leftarrow \lceil \log_2(2N) \rceil$ ▷ Initialize doubling schedule
while `True` **do**
  $\mathcal{I}^{(k)} \leftarrow \texttt{DetectIntervals}\left(\frac{\delta_{\text{explore}}}{4}, 2^k\right)$ ▷ *(i) Detect candidate intervals*
  if $|\mathcal{I}^{(k)}| \geq N$ **then**
    $\left(\mathcal{J}^{(k)}, \mathcal{G}^{(k)}\right) \leftarrow \texttt{EstimateJumps}\left(\mathcal{I}^{(k)}, \frac{\delta_{\text{explore}}}{4}, 2^k, N\right)$ ▷ *(ii) Estimate jump magnitudes*
    if $|\mathcal{J}^{(k)}| \geq N$ **then** ▷ If N intervals detected and estimated
      Order $\mathcal{J}^{(k)} = (J_1^{(k)}, \ldots, J_N^{(k)})$ according to estimate jump from $\mathcal{G}^{(k)}$
      for $v = 1, \ldots, N$ **do**
        $c_v^{(k)} \leftarrow \texttt{SHB}\left(J_v^{(k)}, T_v^{(k)}, \eta\right)$ ▷ *(iii) Refine location to precision η; for $T_v^{(k)}$ see* (6)
        $\text{ok}_v^{(k)} \leftarrow \texttt{VerifyCP}\left(l_v^{(k)}, r_v^{(k)}, \delta^{(k)}, T_v^{(k)}\right)$ ▷ *(iv) Verify estimate; for l_v, r_v see* (7)
      end for
      if $\forall v \in [N]$, $\text{ok}_v^{(k)}$ is `True` **then** ▷ If all change points verified
        return $\mathcal{C} \leftarrow \left\{c_1^{(k)}, \ldots, c_N^{(k)}\right\}$ ▷ Stop and output estimates
      end if
    end if
  end if
  $k \leftarrow k + 1$, and $\delta^{(k)} \leftarrow \frac{3\delta}{2\pi^2 N k^2}$ ▷ Double budget and retry
end while

(i) Detection. By Lemma 1, if $T_k \gtrsim \omega_1 H_{\text{detect}}^{(m)}\big(\log(1/\delta_{\text{explore}}) + \omega_2\big)$, then with probability at least $1 - \delta_{\text{explore}}/4$, `DetectIntervals` returns m disjoint intervals that isolate each of the m change points.
(ii) Jump estimation. On the same event, and with the same budget, Lemma 3 implies that `EstimateJumps` returns N intervals and constant-factor accurate jump estimates. Therefore the allocation weights $\alpha_v^{(k)}$ are within constants of the oracle proportions $\Delta_{(v)}^{-2}/H_{\text{localize}}^{(N)}$.
(iii) Refinement. Given these allocations, Lemma 4 (from Lazzaro and Pike-Burke, 2025b) yields that the v-th change point is localized to accuracy η once it receives budget of order $\Delta_{(v)}^{-2}\log(1/(\delta^{(k)}\eta))$. Summing over $v \in [N]$ gives a sufficient condition for the refinement step to succeed with probability at least $1 - \delta^{(k)}$: $T_k \gtrsim H_{\text{localize}}^{(N)}\big(\log(1/(\delta^{(k)}\eta)) + \omega_3\big)$.
(iv) Verification. By Lemma 5, when $[l_v^{(k)}, r_v^{(k)}]$ contains the v-th change point, with jump magnitude $\Delta_{(v)}$, as long as $T_v^{(k)} \gtrsim \Delta_{(v)}^{-2}\log(1/(\delta^{(k)}\eta))$, `VerifyCP` returns `True` with high probability. Thus, once (i)–(iii) hold, the stage is successful with high probability, and the algorithm stops.
Combining (i)–(iv), a sufficient stage condition, implying both bounds is

$$T_k \gtrsim \omega_1 H_{\text{detect}}^{(m)}\big(\log(1/\delta_{\text{explore}}) + \omega_2\big) + H_{\text{localize}}^{(N)}\big(\log(1/(\delta^{(k)}\eta)) + \omega_3\big) \ .$$

4 Lower bounds

In this section, we derive information-theoretic lower bounds for the budget $\mathcal{T}$ of any (δ, η, m)-correct algorithm for the active change point detection problem in the case where $N = m$.

Consider an environment ν with $m \geq 1$ change points, for which we have to estimate every change point (i.e., $N = m$). As we consider different environments obtained by modification of ν, we use $H_{\text{detect}}^{(m,\nu)}$ (see (2)) and $H_{\text{localize}}^{(m,\nu)}$ (see (1)) to indicate their dependence on the environment ν.

Theorem 2. *Let $\delta \in (0, 1/4)$, $\eta \in (0, 1/8)$, and consider $m = N$. Let π be a (δ, η, m)-correct algorithm. For any environment ν with $m = N$ change points, there exists an environment ν' such*

that $H^{(m,\nu)}_{\text{detect}} \le H^{(m,\nu')}_{\text{detect}} \le 4H^{(m,\nu)}_{\text{detect}}$, $\frac{1}{2}H^{(m,\nu)}_{\text{localize}} \le H^{(m,\nu')}_{\text{localize}} \le H^{(m,\nu)}_{\text{localize}}$, *and such that*

$$\mathbb{P}_{\pi,\nu'}\left(\mathcal{T}_\pi \ge \tfrac{1}{4}H^{(m,\nu)}_{\text{detect}} \log\left(\frac{1}{8\delta}\right) + \tfrac{1}{2}H^{(m,\nu)}_{\text{localize}} \log\left(\frac{1}{8\delta}\right) + \tfrac{1}{2}\sum_{i=1}^{m} \frac{1}{\Delta_i^2} \log_+\left(\frac{s_i}{16\eta}\right)\right) \ge \delta \ .$$

Sketch of proof. The proof is information-theoretic. The high-probability statement combines two ingredients, proved separately in Lemmas 6 and 7.

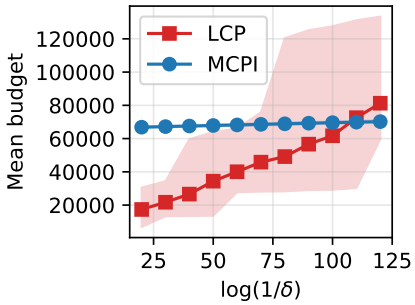
(i) Detection term. To obtain a $(1-\delta)$-quantile lower bound of order $H^{(m,\nu)}_{\text{detect}} \log(1/\delta)$, we reduce the problem to signal detection. This technique is, from our perspective, original and is, in particular, of interest for other active statistical problems. For intuition, consider an instance with two opposite jumps $\pm\Delta$ separated by spacing s, and compare it with a null environment with no change point. If an algorithm detects a change with probability at least $1-\delta$ using budget quantile T, it must sample sufficiently often inside the true plateau, whose location is unknown among about $1/s$ candidates shifts. This yields $T \gtrsim \frac{1}{s\Delta^2}\log\left(\frac{1}{\delta}\right)$, which is exactly the energy-driven detection scaling. Our reduction extends this intuition to the general case. Our reduction requires $N = m$, so that removing one change point from the environment makes any (δ, η, N)-correct algorithm fail to stop.
(ii) Localization at precision η. To obtain the term $\sum \Delta_i^{-2} \log(s_i/\eta)$, we construct a family of alternatives by shifting each change point x_i^* over $\alpha_i \asymp s_i/8\eta$ candidate positions. Distinguishing these alternatives induces the factor $\log_+(s_i/\eta)$ in the quantile bound. We exploit symmetry techniques to reveal this multiple-testing cost. The term $H^{(m,\nu)}_{\text{localize}} \log(1/\delta)$ follows from standard change-of-measure arguments [Lattimore and Szepesvári, 2020, Tsybakov, 2009], as in Theorem 5.2 in Lazzaro and Pike-Burke [2025a].
By standard Markov-type arguments, the high-probability lower bound of Theorem 2 implies the expectation lower bound stated in the following corollary. See Appendix E for the full proof. □

Corollary 1. *Let $\delta \in (0, 1/16)$ and $\eta \in (0, 1/8)$. Assume that all change points of ν are spaced by at least 2η ($\forall i = 2, \dots, m-1,\ \vartheta_i > 2\eta$). Then, there exists an environment ν' such that $H^{(m,\nu)}_{\text{detect}} \le H^{(m,\nu')}_{\text{detect}} \le 4H^{(m,\nu)}_{\text{detect}}$, $\frac{1}{2}H^{(m,\nu)}_{\text{localize}} \le H^{(m,\nu')}_{\text{localize}} \le H^{(m,\nu)}_{\text{localize}}$, and such that*

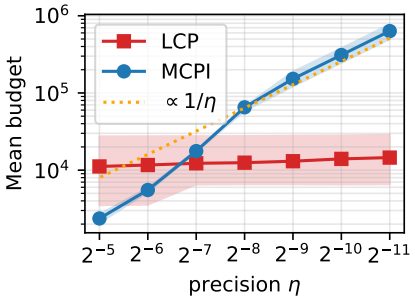
$$\mathbb{E}_{\pi,\nu'}[\mathcal{T}_\pi] \ge c\left(H^{(m,\nu)}_{\text{detect}} + H^{(m,\nu)}_{\text{localize}} \log\left(\frac{1}{4\delta}\right) + \sum_{i=1}^{m} \frac{1}{\Delta_i^2} \log_+\left(\frac{s_i}{16\eta}\right)\right) \ ,$$

where $c > 0$ is a numerical constant.

5 Numerical experiments



(a) **Experiment 2**. Average budget and $(0.05, 0.95)$ quantiles for localizing two CPs with varying δ.



(b) **Experiment 3**. Average budget and $(0.05, 0.95)$ quantiles for localizing two CPs with varying η.

Figure 2: Numerical experiments

Our theoretical findings are supported by simple numerical experiments on simulated data, where we model the underlying noise to be standard normal. In the following two experiments, we consider an environment with two change points, $x_1^* \sim \mathcal{U}(0, 1/2)$ and $x_2^* = x_1^* + s$ with $s = 1/4$, and jump sizes $\Delta_1 = -\Delta_2 = 1$. Each experiment is averaged over 1000 Monte Carlo runs. We compare our method, `LCP`, to `MCPI` from Lazzaro and Pike-Burke [2025a]. Recall that `MCPI` is designed for discrete bandit change point problems, so we apply it to a discretization of the continuous problem (see Appendix F). `MCPI` follows a Track-and-Stop approach with a forced exploration phase. We will see that while the

number of arms, here of order $1/\eta$, does not affect the asymptotic sample complexity as $\delta \to 0$, it strongly influences the algorithm's performance in practice.

In **Experiment 2**, we set $\eta = 2^{-8}$. With these configurations, we run `LCP` (again we treat δ_{explore} as tuning parameter and set $\delta_{\text{explore}} = 1$) and `MCPI` on a discretization for varying values of δ, that is, $\log(1/\delta) \in \{20, 30, \dots, 120\}$. We plot the averaged budget in Figure 2a. We can see that in this setting, the slope of the budget for `LCP` with respect to $\log(1/\delta)$ is larger than the very small slope of the budget required for `MCPI`. Still, it takes $\delta \approx e^{-110}$ until the average budget of our method is larger.

In **Experiment 3**, we choose $\delta = 0.05$ and run `LCP` (again, with $\delta_{\text{explore}} = 1$) and `MCPI` for $\eta = 2^{-i}$, $i = 5, 6, \dots, 11$. We plot the results of our experiment in Figure 2b. For `LCP`, there is only a small increase of the budget when η decreases, compared to `MCPI`. For the latter, we can see that for small values of η, the budget increases nearly proportionally to $1/\eta$.

Both experiments illustrate that for moderate choices of δ, asking for a small precision (or considering many arms in the discrete case) does not limit the performance of `LCP`, in contrast to `MCPI`. The code to reproduce all experiments is available at `bandit_changepoint`. We refer to Appendix C for an evaluation of the runtime and further experiments. Our experiments also suggest that the large constant c in Theorem 1 is an artifact of the proof, as it is not reflected in practice.

6 Discussion

Comparison with Lazzaro and Pike-Burke [2025a]. Lazzaro and Pike-Burke [2025a] study the same problem, but their analysis targets the asymptotic regime $\delta \to 0$ with all other parameters fixed. In this regime the detection stage vanishes from the complexity, and the leading term reduces to $H^{(N)}_{\text{localize}} \log(1/\delta)$, corresponding to N independent localizations of a single change point. This reduction misses a key phenomenon when $m \geq 2$: estimating several closely located jumps induces an additional detection cost, which dominates when δ is bounded away from zero. Our analysis isolates this term as $H^{(m)}_{\text{detect}}$, shows that it enters additively in expectation and does not depend on δ, and provides a matching lower bound (Theorem 2). We also sharpen the dependence on precision: replacing the uniform η-grid initialization of Lazzaro and Pike-Burke [2025a] – which incurs a detection cost of order η^{-1} – with multiscale detection followed by binary-search refinement reduces the η dependence to logarithmic, so the localization component scales as $H^{(N)}_{\text{localize}} \log(1/\eta)$. More generally, the guarantees are non-asymptotic in both (η, δ) and adaptive to local spacing s_i, in contrast to approaches that require a global separation condition $\min_i \vartheta_i > \eta$.

Optimality. When $N = m$, the upper and lower bounds match up to polylogarithmic factors which do not depend on δ and η. Hence `LCP` is rate-optimal across the full range of (δ, η), which characterizes the sample complexity of active multiple change point identification up to these factors. In the limit $\delta \to 0$, our expectation bound scales as $H^{(N)}_{\text{localize}} \log(1/\delta)$, confirming rate-optimality. `MCPI` [Lazzaro and Pike-Burke, 2025a] attains the exact optimal constant $\lim_{\delta \to 0} \mathbb{E}[\mathcal{T}]/\log(1/\delta) = 8H^{(N)}_{\text{localize}}$, while our bound is order-optimal with a looser constant; closing this gap may require a generalized likelihood ratio stopping rule, as in Track-and-Stop methods [Garivier and Kaufmann, 2016].

Fixed budget setting. Our procedure targets the fixed-confidence setting, but one stage of the doubling schedule in Algorithm 2 could potentially be adapted to fixed-budget. Doing so would require a parameter-free exploration phase, whereas the current method relies on a tuning parameter δ_{explore} to control failure probability. Such a fixed-budget variant would also address the case $N > m$, where the fixed-confidence formulation is ill-posed: any algorithm targeting N distinct change points would loop indefinitely with probability at least $1 - \delta$. A related alternative is to replace the input N by a minimal energy threshold $E_{\min}$, targeting all change points with energy at least $E_{\min}$; we discuss this extension in Appendix B. We leave both directions for future work.

Extensions. The detection–localization decomposition is modular: the two-sample tests and the binary search can be replaced by richer procedures, suggesting extensions to smoother signal classes, vector-valued outputs, and multi-dimensional input spaces (Appendix B).

Limitations. Our lower bounds are limited to the case $N = m$. When $N < m$, the upper bound is likely loose, and the optimal trade-off remains a highly non-trivial open question; we discuss this regime and a conjecture in Appendix B. Algorithmically, our procedure relies on a doubling schedule, which is not always desirable in practice. Adaptations are required to derive a fully-adaptive counterpart of the method.

Acknowledgments and Disclosure of Funding

The work of M. Graf is partially supported by the Deutsche Forschungsgemeinschaft (DFG) – Project-ID 318763901 – SFB1294 "Data Assimilation", Project A03, by the DFG within the Forschungsgruppe FOR5381 "Mathematical Statistics in the Information Age – Statistical Efficiency and Computational Tractability", Project TP 02 (Project-ID 460867398), and by the Université franco-allemande (UFA) through the Collège doctoral franco-allemand CDFA-02-25 "Statistisches Lernen für komplexe stochastische Prozesse". The work of V. Thuot has partially been supported by ANR-21-CE23-0035 (ASCAI, ANR).

References

Tze Leung Lai. Sequential analysis: some classical problems and new challenges. *Statistica Sinica*, pages 303–351, 2001.

Yi Yu. A review on minimax rates in change point detection and localisation. *arXiv preprint arXiv:2011.01857*, 2020.

Charles Truong, Laurent Oudre, and Nicolas Vayatis. Selective review of offline change point detection methods. *Signal processing*, 167:107299, 2020.

Pavel Nikolaev, Daylond Hooper, Frederick Webber, Rahul Rao, Kevin Decker, Michael Krein, Jason Poleski, Rick Barto, and Benji Maruyama. Autonomy in materials research: a case study in carbon nanotube growth. *npj Computational Materials*, 2(1):1–6, 2016.

Nerea Sebastián, MR De La Fuente, DO López, MA Pérez-Jubindo, J Salud, S Diez-Berart, and MB Ros. Dielectric and thermodynamic study on the liquid crystal dimer α-(4-cyanobiphenyl-4'-oxy)-ω-(1-pyreniminebenzylidene-4'-oxy) undecane (cbo11o· py). *The Journal of Physical Chemistry B*, 115(32):9766–9775, 2011.

YC Hung and G Michailidis. Modeling, analysis and simulation of switched processing systems. *ACM Transactions on Modeling, Analysis and Simulation*, 2007.

Yan Lan, Moulinath Banerjee, and George Michailidis. Change-point estimation under adaptive sampling. *The Annals of Statistics*, 37(4):1752–1791, 2009.

Shogo Hayashi, Yoshinobu Kawahara, and Hisashi Kashima. Active change-point detection. In *Asian Conference on Machine Learning*, pages 1017–1032. PMLR, 2019.

Tor Lattimore and Csaba Szepesvári. *Bandit Algorithms*. Cambridge University Press, 2020.

Joseph Lazzaro and Ciara Pike-Burke. Fixed-confidence multiple change point identification under bandit feedback. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 32675–32703. PMLR, 13–19 Jul 2025a. URL `https://proceedings.mlr.press/v267/lazzaro25a.html`.

Yue S Niu, Ning Hao, and Heping Zhang. Multiple change-point detection: a selective overview. *Statistical Science*, pages 611–623, 2016.

Nicolas Verzelen, Magalie Fromont, Matthieu Lerasle, and Patricia Reynaud-Bouret. Optimal change-point detection and localization. *The Annals of Statistics*, 51(4):1586–1610, 2023.

Joe Suk and Samory Kpotufe. Tracking most significant arm switches in bandits. In *Conference on Learning Theory*, pages 2160–2182. PMLR, 2022.

Anne Gael Manegueu, Alexandra Carpentier, and Yi Yu. Generalized non-stationary bandits. *arXiv preprint arXiv:2102.00725*, 2021.

Aurélien Garivier and Emilie Kaufmann. Optimal best arm identification with fixed confidence. In *Conference on Learning Theory*, pages 998–1027. PMLR, 2016.

David V Hinkley. Inference about the change-point in a sequence of random variables. *Biometrika*, pages 1–17, 1970.

A.J. Scott and Martin Knott. A cluster analysis method for grouping means in the analysis of variance. *Biometrics*, pages 507–512, 1974.

Piotr Fryzlewicz. Wild binary segmentation for multiple change-point detection. *The Annals of Statistics*, 42(6):2243–2281, 2014.

Daren Wang, Yi Yu, and Alessandro Rinaldo. Univariate mean change point detection: Penalization, cusum and optimality. *Electronic Journal of Statistics*, 14:1917–1961, 2020.

Solt Kovács, Peter Bühlmann, Housen Li, and Axel Munk. Seeded binary segmentation: a general methodology for fast and optimal changepoint detection. *Biometrika*, 110(1):249–256, 2023.

Emmanuel Pilliat, Alexandra Carpentier, and Nicolas Verzelen. Optimal multiple change-point detection for high-dimensional data. *Electronic Journal of Statistics*, 17(1):1240–1315, 2023.

Tengyao Wang and Richard J Samworth. High dimensional change point estimation via sparse projection. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 80(1):57–83, 2018.

Sylvain Arlot, Alain Celisse, and Zaid Harchaoui. A kernel multiple change-point algorithm via model selection. *Journal of machine learning research*, 20(162):1–56, 2019.

Changliang Zou, Guosheng Yin, Long Feng, and Zhaojun Wang. Nonparametric maximum likelihood approach to multiple change-point problems. *The Annals of Statistics*, 42(3):970–1002, 2014. ISSN 0090-5364.

Rebecca Killick, Paul Fearnhead, and Idris A Eckley. Optimal detection of changepoints with a linear computational cost. *Journal of the American Statistical Association*, 107(500):1590–1598, 2012.

Imon Banerjee, Jiaqi Lei, and Sanjay Mehrotra. Nonparametric multi change point detection for markov chains via adaptive clustering. In *The 29th International Conference on Artificial Intelligence and Statistics*, 2026.

Aurélien Garivier and Eric Moulines. On upper-confidence bound policies for switching bandit problems. In *International conference on algorithmic learning theory*, pages 174–188. Springer, 2011.

Venugopal V Veeravalli, Georgios Fellouris, and George V Moustakides. Quickest change detection with controlled sensing. *IEEE Journal on Selected Areas in Information Theory*, 5:1–11, 2024.

Wanrong Zhang and Yajun Mei. Bandit change-point detection for real-time monitoring high-dimensional data under sampling control. *Technometrics*, 65(1):33–43, 2023.

Qunzhi Xu and Yajun Mei. Asymptotic optimality theory for active quickest detection with unknown postchange parameters. *Sequential analysis*, 42(2):150–181, 2023.

Yang Cao, Zheng Wen, Branislav Kveton, and Yao Xie. Nearly optimal adaptive procedure with change detection for piecewise-stationary bandit. In *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89 of *Proceedings of Machine Learning Research*, pages 418–427. PMLR, 16–18 Apr 2019. URL `https://proceedings.mlr.press/v89/cao19a.html`.

Lilian Besson, Emilie Kaufmann, Odalric-Ambrym Maillard, and Julien Seznec. Efficient change-point detection for tackling piecewise-stationary bandits. *Journal of Machine Learning Research*, 23(77):1–40, 2022.

Subhojyoti Mukherjee. Safety aware changepoint detection for piecewise i.i.d. bandits. In *The 38th Conference on Uncertainty in Artificial Intelligence*, 2022. URL `https://openreview.net/forum?id=rgZGGvLi5eq`.

Richard Combes and Alexandre Proutiere. Unimodal bandits: Regret lower bounds and optimal algorithms. In *International Conference on Machine Learning*, pages 521–529. PMLR, 2014.

Wataru Kumagai. Regret analysis for continuous dueling bandit. *Advances in Neural Information Processing Systems*, 30, 2017.

Sébastien Bubeck, Rémi Munos, and Gilles Stoltz. Pure exploration in finitely-armed and continuous-armed bandits. *Theoretical Computer Science*, 412(19):1832–1852, 2011.

Peter Hall and Ilya Molchanov. Sequential methods for design-adaptive estimation of discontinuities in regression curves and surfaces. *The Annals of Statistics*, 31(3):921–941, 2003.

Joseph Lazzaro and Ciara Pike-Burke. Fixed-budget change point identification in piecewise constant bandits. In *The 28th International Conference on Artificial Intelligence and Statistics*, 2025b. URL `https://openreview.net/forum?id=g8DVAMyn1j`.

James Cheshire, Pierre Ménard, and Alexandra Carpentier. The influence of shape constraints on the thresholding bandit problem. In *Conference on Learning Theory*, pages 1228–1275. PMLR, 2020.

Junwen Yang, Zixin Zhong, and Vincent YF Tan. Optimal clustering with bandit feedback. *Journal of Machine Learning Research*, 25(186):1–54, 2024.

Maximilian Graf, Victor Thuot, and Nicolas Verzelen. Clustering items through bandit feedback: Finding the right feature out of many. In *International Conference on Machine Learning*, pages 20296–20325. PMLR, 2025.

Alexandre B. Tsybakov. *Introduction to nonparametric estimation*. Springer Series in Statistics. Springer, 2009.

James Cheshire, Pierre Ménard, and Alexandra Carpentier. Problem dependent view on structured thresholding bandit problems. In *International Conference on Machine Learning*, pages 1846–1854. PMLR, 2021.

Oscar Hernan Madrid Padilla, Yi Yu, and Alessandro Rinaldo. Lattice partition recovery with dyadic cart. *Advances in Neural Information Processing Systems*, 34:26143–26155, 2021.

Sébastian Bubeck, Tengyao Wang, and Nitin Viswanathan. Multiple identifications in multi-armed bandits. In *International Conference on Machine Learning*, pages 258–265. PMLR, 2013.

Shivaram Kalyanakrishnan, Ambuj Tewari, Peter Auer, and Peter Stone. Pac subset selection in stochastic multi-armed bandits. In *Proceedings of the 29th International Conference on Machine Learning, ICML 2012*, volume 12, pages 655–662. PMLR, 01 2012.

Yury Polyanskiy and Yihong Wu. Lecture notes on information theory. *Lecture Notes for ECE563 (UIUC) and*, 6(2012-2016):7, 2014.

A List of notation

We summarize the notation used throughout the paper in Table 1.

Table 1: Summary of notation.

Symbol	Description
f	Unknown step function
m	Total number of change points
x_i^*	Location of the i-th change point
x_t	Location, chosen by a learner at time t
ε_t	SubGaussian noise at time t
y_t	Feedback $y_t = f(x_t) + \varepsilon_t$ that the learner observes at time t
N	Number of change points the learner wants to localize
η	Precision parameter
δ	Confidence parameter for correctness
δ_{explore}	Confidence parameter for exploration
$\mathcal{T}$	Sample complexity / budget
Δ_i	Jump at the i-th change point, $\Delta_i = f(x_i^*) - f(x_{i-1}^*)$
$\Delta_{(i)}$	Jump ordered by its magnitude such that $\lvert\Delta_{(1)}\rvert \geq \cdots \geq \lvert\Delta_{(m)}\rvert$
ϑ_i	Spacing after the i-th change point, $\vartheta_i = x_{i+1}^* - x_i^*$ for $i = 1, \ldots, m-1$, with convention $\vartheta_0 = \vartheta_m = 1$
s_i	Local spacing between the i-th change point and its neighboring change points, $s_i = \vartheta_{i-1} \wedge \vartheta_i$ for $i = 1, \ldots, m$,
$\mathcal{E}_i^2$	Energy of the i-th change point, $\mathcal{E}_i^2 = s_i \Delta_i^2$
$H_{\text{detect}}^{(m)}$	Complexity term for change point detection, $H_{\text{detect}}^{(m)} = \max_{i=1,\ldots,m} \mathcal{E}_i^{-2}$
$H_{\text{localize}}^{(N)}$	Complexity term for change point localization, $H_{\text{localize}}^{(N)} = \sum_{i=1}^{N} 1/\Delta_{(i)}^2$
$l(I), r(I)$	Boundaries of a compact interval $I = [l(I), r(I)]$
$\Delta(I)$	Jump between interval boundaries, $\Delta(I) = f(r(I)) - f(l(I))$

B Extensions and perspectives

We collect here open problems within the current framework and directions for extending it to richer settings.

Optimal complexity when $N < m$. When $N < m$, the upper bound is likely loose: $H_{\text{detect}}^{(m)}$ accounts for all m change points, so low-energy untargeted change points can inflate the detection cost without affecting localization. Consider $N = 2$, $m = 4$, with untargeted change points at $x_1^* = \frac{1}{4} - \varepsilon$, $x_4^* = \frac{3}{4} + \varepsilon$ and targeted ones at $x_2^* = \frac{1}{4}$, $x_3^* = \frac{3}{4}$. Setting $\Delta_1 = \varepsilon = -\Delta_4$ and $\Delta_2 = 1 = -\Delta_3$ gives $\mathcal{E}_1^2 = \mathcal{E}_4^2 = \varepsilon^3$ and $\mathcal{E}_2^2 = \mathcal{E}_3^2 = \varepsilon$, so $H_{\text{detect}}^{(4)} \asymp \varepsilon^{-3} \to \infty$ while $H_{\text{localize}}^{(2)} = 2$. Yet LCP with $N = 2$ should not pay this price: at coarse depths the jump across $[0, \frac{1}{4}]$ is $\Delta_1 + \Delta_2 \approx 1$, so this interval is detected at $O(1)$ cost and the perturbation ε is never resolved. The relevant detection cost should therefore involve the energy of *detectable* change points – a notion that differs from s_i when $N < m$, and explains why $H_{\text{detect}}^{(m)}$ overestimates the true cost. Formalizing this and deriving matching lower bounds is left for future work.

Energy-based identification. The (δ, η, N)-correctness framework requires $N \leq m$; if $N > m$, no algorithm terminates with high probability under fixed confidence, since the smallest energy $\mathcal{E}_i$ is

unknown and the algorithm cannot rule out the existence of a change point with arbitrarily low energy. A natural alternative is to fix a minimal energy threshold $E_{\min}$ and ask the algorithm to identify *all* change points with energy $\mathcal{E}_i \geq E_{\min}$, within precision η and with confidence $1-\delta$; the total number of such change points is then recovered as a by-product. We expect that our methodology can be adapted to this setting. From Lemma 1, a budget $T \gtrsim E_{\min}^{-2} \log(1/\delta)$ suffices to detect all change points with energy at least $E_{\min}$ with probability at least $1-\delta$. Combining this with the refinement and verification steps yields a sufficient budget of

$$T \;\gtrsim\; \frac{1}{E_{\min}^2} \log\left(\frac{1}{\delta}\right) + \sum_{i:\, \mathcal{E}_i \geq E_{\min}} \frac{1}{\Delta_i^2} \log\left(\frac{s_i}{\eta}\right),$$

up to logarithmic factors. This framework still requires knowledge of a lower bound on the energy of the change points to be detected, mirroring the role of $N \leq m$ in the original setting. We leave a full treatment for future work.

The algorithmic structure of `LCP` – detecting rough regions that contain a change, then refining their location – is not specific to the one-dimensional piecewise-constant model. We sketch here how it can be adapted to several richer settings.

Beyond piecewise-constant: significant changes. Suk and Kpotufe [2022] introduce the notion of *significant arm switches* to avoid the piecewise-constant assumption in non-stationary bandits. Transposed to our spatial setting, this amounts to replacing the definition of change points by that of *significant changes*: points $\tau \in [0,1]$ at which the local energy

$$\mathcal{E}_{\tau,s} = \left| \frac{\int_{\tau}^{\tau+s} f(x)\,\mathrm{d}x - \int_{\tau-s}^{\tau} f(x)\,\mathrm{d}x}{\sqrt{s}} \right|$$

is large for some scale $s > 0$, together with an appropriate hierarchical pruning to remove redundant detections [Suk and Kpotufe, 2022]. This formulation is general: it covers in particular piecewise-linear and piecewise-polynomial functions [Manegueu et al., 2021]. We expect that `LCP` can be adapted to this setting, using the definitions and reduction tools of Suk and Kpotufe [2022] to relate it back to the piecewise-constant change point scenario, possibly without requiring new theoretical arguments beyond those developed here.

Smooth transitions. The piecewise-constant model assumes instantaneous jumps, which may be idealized in practice. Our method can be adapted to settings where the target locations are smooth transitions between levels, by replacing the two-sample endpoint test with a threshold-crossing criterion, as in Cheshire et al. [2021]. Binary search remains applicable in this context: it achieves precision η in $O(\log(1/\eta))$ queries, compared to $O(1/\eta)$ for a uniform sweep – an exponential gain that our algorithm extends to multiple transitions through the same detection-then-localization decomposition.

Piecewise-linear and piecewise-polynomial functions. In a piecewise-linear model, the change points are discontinuities of f' rather than of f. Our method can be adapted by estimating f' from finite differences: querying $f(x)$ and $f(x+\varepsilon)$ for small ε yields derivative estimates that replace raw values in the two-sample tests used by `DetectIntervals` and `VerifyCP`. The overall algorithmic structure is unchanged, and we expect the main guarantees to extend, with the jump magnitudes Δ_i replaced by the corresponding derivative jumps. The same principle applies to piecewise-polynomial models of any fixed degree, by targeting discontinuities of higher-order finite differences.

Vector-valued outputs. Extending to vector-valued functions $f : [0,1] \to \mathbb{R}^d$ mainly requires replacing the scalar two-sample test with an appropriate multivariate procedure [Wang and Samworth, 2018], possibly sparse-adaptive when d is large [Pilliat et al., 2023]. The detection-localization decomposition and the doubling schedule carry over without modification. In this case, remark that the computational cost in higher dimension is linear in that of a single high-dimensional two-sample test.

Multi-dimensional input space. Moving to $f : [0,1]^d \to \mathbb{R}$ with $d \geq 2$ changes the nature of the problem substantially: one now seeks a *boundary* rather than isolated points, and the learning

objective admits several non-equivalent formulations. This setting is still far from being fully understood even in the passive literature. Nevertheless, the core conceptual contribution of our work – a multistage procedure that first identifies coarse regions containing a change and then refines their location – may remain applicable. For instance, it can potentially be combined with the dyadic CART framework of Madrid Padilla et al. [2021], where the change boundary is a rectangular partition of $[0,1]^d$. We leave a formal treatment of this direction for future work.

C Further numerical experiments and runtime analysis

All experiments were run on a MacBook Air (Apple M1, 8 GB RAM), using 4 parallel worker processes. The runtime is documented in table 2.

Table 2: Computational resources used for each experiment (runtimes in sec.)

	# Monte Carlo runs	**Total runtime**	**Runtime** `LCP`	**Runtime** `MCPI`
Experiment 1	1000	6667.64	433.76	26277.82
Experiment 2	1000	1035.69	145.15	3987.08
Experiment 3	1000	1634.18	37.19	6487.09
Experiment 4	1000	599.06	37.31	2354.73
Experiment 5	100	2619.28	163.92	10281.14

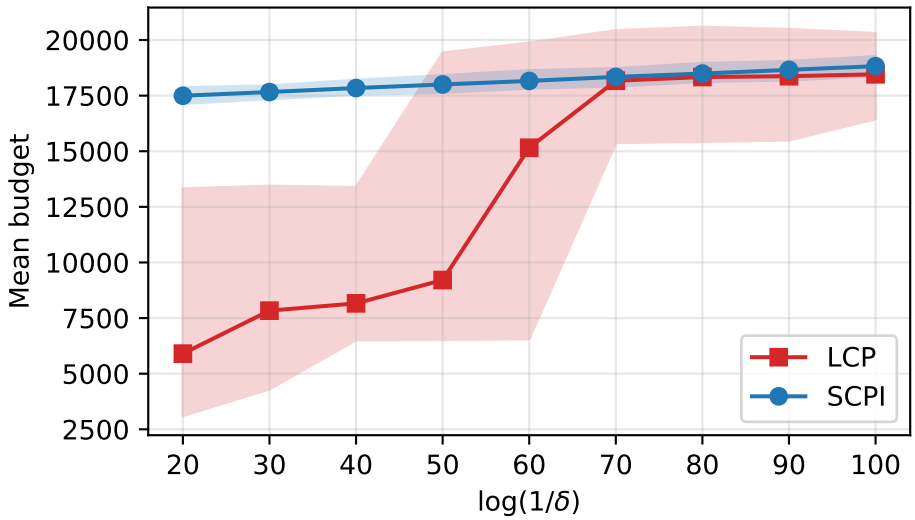


Figure 3: **Experiment 4**. Average budget for localizing one change point with varying δ.

We illustrate **Experiment 4** in Figure 3. We consider one change point x_1^* and $\Delta_1 = 1$. Again, we consider standard normal noise. In 1000 Monte Carlo runs, we run `LCP` (with $\delta_{\text{explore}} = 1$) and `SCPI` with precision $\eta = 2^{-7}$ and varying δ. `SCPI` (*Single Change Point Identification*) is an adaptation of `MCPI`, also provided in Lazzaro and Pike-Burke [2025a] for identification of a single change point. Like in Experiment 2, we can see that while the budget of our method increases stronger for $\log(1/\delta)$ growing, we still achieve better performance for moderate choices of δ.

In **Experiment 5**, illustrated in Figure 4, we consider a problem with $m = 10$ change points and $\Delta_i = 1$ for i odd and $\Delta_i = -1$ for i even. The noise is again supposed to be standard normal. We run `MCPI` and `LCP` (with $\delta_{\text{explore}} = 1$) with varying $\log(1/\delta) \in \{20, 40, 60, 80, 100\}$ and $\eta = 0.0025 \cdot 2^{-i}$, $i = 1, 2, 3$. For `LCP` we observe that only the choice of δ has a visible but mild effect on the average budget. For `MCPI`, increasing η by a factor 4 leads to a decrease of the budget by a factor $1/4$. Meanwhile, the slope seems to depend almost linearly on $\log(1/\delta)$.

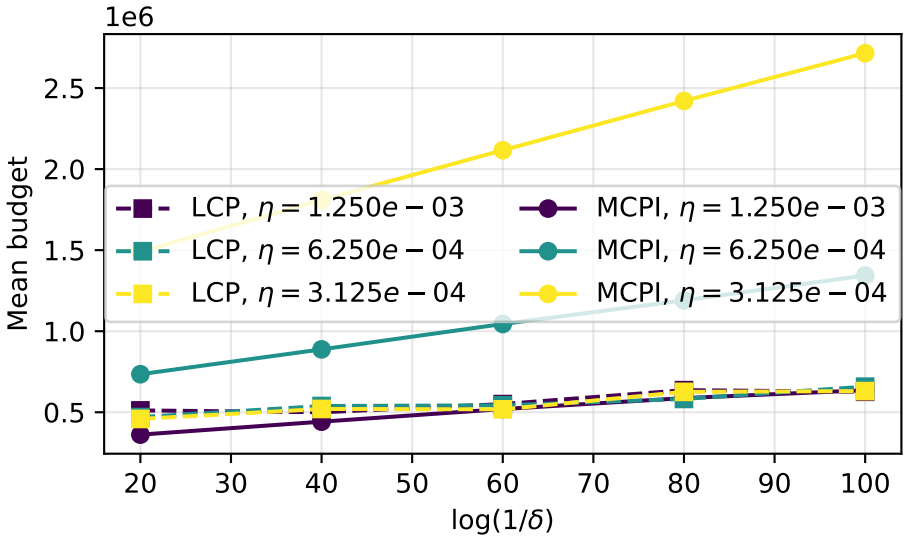


Figure 4: **Experiment 5**. Average budget for localizing 10 change points for three values of η and varying δ.

D Proofs of upper bounds

D.1 Multiple change point detection

We now prove Lemma 1, which is stated in Section 3.1.

Proof of Lemma 1. We refer to Section 3.1 and Algorithm 1 for the introduction of the notation used in this proof. Fix one depth $d \in \{1, \dots, d_{\max}\}$ and one index $j \in \{0, \dots, n_d\}$. Let $y_j^{(d)} = f(x_j^{(d)})$ be the true function value at $x_j^{(d)}$. By assumption, the noise is 1-subGaussian, so Hoeffding's inequality yields, for any $\varepsilon > 0$,

$$\mathbb{P}\left(|\hat{y}_j^{(d)} - y_j^{(d)}| \geq \varepsilon\right) \leq 2\exp\left(-\frac{T_d\varepsilon^2}{2}\right) \quad .$$

From a union bound, first over all $n_d + 1$ endpoints of depth d, and then over each depth $d = 1, \dots, d_{\max}$, we obtain an event ξ with $\mathbb{P}(\xi) \geq 1 - \delta$ such that for all $d = 1, \dots, d_{\max}$ and all $j = 0, \dots, n_d$,

$$|\hat{y}_j^{(d)} - y_j^{(d)}| \leq \sqrt{\frac{2}{T_d}\log\left(\frac{2d_{\max}(n_d+1)}{\delta}\right)} = \beta_d/2. \tag{8}$$

First, on ξ, intervals without change points are not added to $\mathcal{I}$. Indeed, consider any depth d and any interval $[x_{j-1}^{(d)}, x_j^{(d)}]$ that does not contain a change point. Then $y_{j-1}^{(d)} = y_j^{(d)}$, and therefore, on ξ (see (8)),

$$|\hat{y}_j^{(d)} - \hat{y}_{j-1}^{(d)}| \leq |\hat{y}_j^{(d)} - y_j^{(d)}| + |y_{j-1}^{(d)} - \hat{y}_{j-1}^{(d)}| \leq \beta_d \quad ,$$

and the interval is not added to $\mathcal{I}$ (Line 8). This guarantees that every interval in $\mathcal{I}$ contains at least one change point, which proves the first part of the statement.

Second, on ξ, change points are detected if the budget is large enough. Fix any change point x_i^*, with $i \in \{1, \dots, m\}$, jump size Δ_i, and minimal spacing s_i to the other change points. Recall that $d_{\max} = \lfloor \log_2(T/\log(1/\delta)) \rfloor$, $n_d = 2^d$, and define the depth specific to change point i

$$d_i^* = \left\lceil \log_2\left(\frac{2}{s_i}\right) \right\rceil \leq d_{\max} \tag{9}$$

The latter inequality ensures that depth d_i^* is considered by the algorithm. It follows from

$$T \geq 2e \cdot \mathcal{E}_i^{-2} \cdot \log(1/\delta) \geq \frac{2e \cdot \log(1/\delta)}{s_i} \ ,$$

recalling that $\mathcal{E}_i^2 = s_i \Delta_i^2 \leq s_i$.

Consider depth d_i^* and the index j such that $x_{j-1}^{(d_i^*)} < x_i^* \leq x_j^{(d_i^*)}$. By (9), we have $2^{-d_i^*} \leq s_i/2$, so the definition of s_i ensures that no other change point is contained in $[x_{j-1}^{(d_i^*)}, x_j^{(d_i^*)}]$, and $y_j^{(d_i^*)} - y_{j-1}^{(d_i^*)} = \Delta_i$. Moreover, on ξ, using (8) and the reverse triangle inequality,

$$|\hat{y}_j^{(d_i^*)} - \hat{y}_{j-1}^{(d_i^*)}| \geq |\Delta_i| - \beta_{d_i^*} \ .$$

Now, if T is large enough to ensure $\Delta_i \geq 2\beta_{d_i^*}$, then the test in Line 8 of Algorithm 1 is positive, and $[x_{j-1}^{(d_i^*)}, x_j^{(d_i^*)}]$ is added to $\mathcal{I}$ during the iteration at depth d_i^*. In particular, x_i^* is contained in an interval of $\mathcal{I}$ with length at most $2^{-d_i^*} \leq s_i/2$. Besides, at the following iterations at depth $d > d_i^*$, the algorithm maintains in $\mathcal{I}$ exactly one interval containing x_i^*, and whose length is at most $s_i/2$.

Indeed, consider any depth $d > d_i^*$ and an interval $[x_{j-1}^{(d)}, x_j^{(d)}]$ included in $[x_{j-1}^{(d_i^*)}, x_j^{(d_i^*)}]$. If it does not contain a change point, it is not added to $\mathcal{I}$, as the event ξ holds (Equation (8)). Otherwise, $[x_{j-1}^{(d_i^*)}, x_j^{(d_i^*)}]$ contains x_i^* (as it is the unique change point in that interval). It may be added to $\mathcal{I}$, in which case the interval from $\mathcal{I}$ containing x_i^* is removed. Therefore, at the end of Algorithm 1, x_i^* is contained in exactly one interval of $\mathcal{I}$, with length at most $2^{-d_i^*} \leq s_i/2$. Moreover, since f is right-continuous, an interval $[x_{j-1}^{(d)}, x_j^{(d)}]$ with no change point in $(x_{j-1}^{(d)}, x_j^{(d)}]$ satisfies $y_{j-1}^{(d)} = y_j^{(d)}$ and is therefore not added to $\mathcal{I}$ on ξ. Hence the interval $I \in \mathcal{I}$ containing x_i^* satisfies $l(I) < x_i^* \leq r(I)$, where $x_i^* = r(I)$ is possible. Since $|I| \leq s_i/2$ and $s_i \leq \vartheta_{i-1} \wedge \vartheta_i$, we further get $l(I) \geq x_i^* - s_i/2 > x_{i-1}^*$ and $r(I) < x_i^* + s_i/2 < x_{i+1}^*$.

It remains to determine a condition on T that guarantees $\Delta_i \geq 2\beta_{d_i^*}$.

Intuitively, by definition of d_i^*, we have $T_{d_i^*} \underset{\log}{\simeq} s_i T$ up to logarithmic factors, so $\Delta_i \geq 2\beta_{d_i^*}$ is roughly equivalent to $T \geq c\, \mathcal{E}_i^{-2} \log(1/\delta)$ up to logarithmic factors. The precise formulation is given in technical Lemma 2, which concludes the proof. □

Lemma 2. *Let* $0 < \delta < 1/4$, $\Delta_i \leq 1$. *If*

$$T \geq 7.4 \cdot 10^5 \cdot (\log(\mathcal{E}_i^{-2}) \vee 1) \cdot \log\left(\frac{\log(\mathcal{E}_i^{-2}) \vee 1}{s_i \cdot \delta}\right) \cdot \mathcal{E}_i^{-2} \ ,$$

then $T_{d_i^*} \geq 1$, *and* $\Delta_i \geq 2\beta_{d_i^*}$.

Proof of Lemma 2. Assume that $T \geq c \cdot (\log(\mathcal{E}_i^{-2}) \vee 1) \cdot \log\left(\frac{\log(\mathcal{E}_i^{-2}) \vee 1}{s_i \cdot \delta}\right) \cdot \mathcal{E}_i^{-2}$ for a constant c we want to determine.

First, we want to verify that $T_{d_i^*} \geq 1$. By definition of the budget allocated to depth d_i^*, we have

$$T_{d_i^*} = \left\lfloor \frac{T}{d_{\max}(n_{d_i^*} + 1)} \right\rfloor \ .$$

We have (since $\delta < 1/4$)

$$d_{\max} \leq \log_2(T/\log(1/\delta)) \leq \frac{\log(T)}{\log(2)}$$

and

$$n_{d_i^*} + 1 = 2^{\left\lceil \log_2\left(\frac{2}{s_i}\right)\right\rceil} + 1 \leq \frac{5}{s_i} \ ,$$

so

$$\frac{d_{\max}(n_{d^*}+1)}{T} \leq \frac{5}{\log(2)} \cdot \frac{\log(T)}{s_i \cdot T} \ .$$

Note that $s_i \leq 1/2$ (otherwise, we could simply pick any $T \geq 60$, which could be relevant in the cases $i = 1$ and $i = N$). Observe that $\mathcal{E}_i^2 = s_i \Delta_i^2 \leq s_i$, so that by assumption on T, we have

$$T \geq c_1 \cdot \frac{\log\left(\frac{1}{s_i}\right)}{s_i}. \tag{10}$$

Then we obtain from $\log(x)/x$ being decreasing for $x \geq e$ that holds

$$\begin{aligned}\frac{5}{\log(2)} \cdot \frac{\log(T)}{s_i \cdot T} &\leq \frac{5}{\log(2)} \cdot \frac{\log(c_1) + \log\left(\frac{1}{s_i}\right) + \log\log\left(\frac{1}{s_i}\right)}{c_1 \log\left(\frac{1}{s_i}\right)} \\ &\leq \frac{5}{\log(2)^2} \cdot \frac{\log(c_1)+2}{c_1} \leq 1 \ ,\end{aligned}$$

where the last inequality holds for $c_1 \geq 65$.

Now, if $c_1 \geq 65$, then $T_{d_i^*} \geq 1$, in which case $\beta_{d_i^*}$ is well-defined. Then, $\lfloor x \rfloor \geq x/2$ for $x \geq 1$, we can therefore obtain

$$\frac{32}{T_{d^*}} \leq 64 \frac{d_{\max}(n_{d^*}+1)}{T} \leq \frac{320}{\log(2)} \cdot \frac{\log(T/\log(1/\delta))}{s_i \cdot T}$$

The objective $\Delta_i \geq 2\beta_{d_i^*}$ is equivalent to the condition

$$4\beta_{d_i^*}^2 = \frac{32}{T_{d_i^*}} \log\left(\frac{2 d_{\max}(n_{d_i^*}+1)}{\delta}\right) \leq \Delta_i^2 \ ,$$

which holds as long as

$$\frac{320}{\log(2)} \cdot \frac{\log(T/\log(1/\delta)) \cdot \mathcal{E}_i^{-2}}{T} \cdot \log\left(\frac{10\log(T/\log(1/\delta))}{\log(2) \cdot s_i \cdot \delta}\right) \leq 1 \ .$$

Note that since $\delta < \frac{1}{4}$, we have

$$\begin{aligned}&\frac{320}{\log(2)} \cdot \frac{\log(T/\log(1/\delta)) \cdot \mathcal{E}_i^{-2}}{T} \cdot \log\left(\frac{10\log(T/\log(1/\delta))}{\log(2) \cdot s_i \cdot \delta}\right) \\ \leq &\frac{320}{\log(2)} \cdot \frac{\log(T/\log(1/\delta)) \cdot \mathcal{E}_i^{-2}}{T} \\ &\quad \times \left[\left(\frac{5}{\log(2)^2}+1\right)\log(1/\delta) + \log\left(\frac{1}{s_i}\right) + \log\log(T/\log(1/\delta))\right] \ .\end{aligned}$$

Assume first, that $\mathcal{E}_i^{-2} \geq e$. We want to determine $c_2, c_3, c_4 > 0$ such that

$$\frac{1600 + 320\log(2)^2}{\log(2)^3} \cdot \frac{\log(T/\log(1/\delta)) \cdot \mathcal{E}_i^{-2}}{T} \cdot \log(1/\delta) \leq 1/3 \tag{11}$$
$$\text{for } T \geq c_2 \cdot \mathcal{E}_i^{-2} \cdot \log(\mathcal{E}_i^{-2}) \cdot \log(1/\delta) \ ;$$

and

$$\frac{320}{\log(2)} \cdot \frac{\log(T/\log(1/\delta)) \cdot \mathcal{E}_i^{-2}}{T} \cdot \log\left(\frac{1}{s_i}\right) \leq 1/3 \tag{12}$$
$$\text{for } T \geq c_3 \cdot \mathcal{E}_i^{-2} \cdot \log(\mathcal{E}_i^{-2}) \cdot \log\left(\frac{1}{s_i}\right) \cdot \log\left(e \vee \log\left(\frac{1}{s_i}\right)\right) \ ;$$

and

$$\frac{320}{\log(2)} \cdot \frac{\log(T/\log(1/\delta)) \cdot \mathcal{E}_i^{-2}}{T} \cdot \log\log(T/\log(1/\delta)) \leq 1/3 \tag{13}$$
$$\text{for } T \geq c_4 \cdot \mathcal{E}_i^{-2} \cdot \log(\mathcal{E}_i^{-2}) \cdot \log(\log(\mathcal{E}_i^{-2}) \vee e) \ .$$

By the monotonicity of $\log(x)/x$ for $x \geq e$, bounding T in inequality (11) yields

$$\begin{aligned}\frac{\log(T/\log(1/\delta)) \cdot \mathcal{E}_i^{-2}}{T} \cdot \log(1/\delta) &\leq \frac{\log(c_2) + \log(\mathcal{E}_i^{-2}) + \log\log(\mathcal{E}_i^{-2})}{c_2 \cdot \log(\mathcal{E}_i^{-2})} \\ &\leq \frac{\log(c_2) + 2}{c_2} \leq \frac{\log(2)^3}{4800 + 960\log(2)^2}\end{aligned}$$

for $c_2 \geq 2.3 \cdot 10^5$.

A similar approach for inequality (12) yields

$$\begin{aligned}&\frac{\log(T/\log(1/\delta)) \cdot \mathcal{E}_i^{-2}}{T} \cdot \log\left(\frac{1}{s_i}\right) \\ \leq \frac{\log(c_3) + \log(\mathcal{E}_i^{-2}) + \log\log(\mathcal{E}_i^{-2}) + \log\log\left(\frac{1}{s_i}\right) + \log\log\left(e \vee \log\left(\frac{1}{s_i}\right)\right)}{c_3 \cdot \log(\mathcal{E}_i^{-2}) \cdot \log\left(e \vee \log\left(\frac{1}{s_i}\right)\right)} \\ &\leq \frac{\log(c_3) + 4}{c_3} \leq \frac{\log(2)}{960}\end{aligned}$$

for $c_3 \geq 2 \cdot 10^4$.

In the same way, we obtain for inequality (13), using

$$\log\log(xy) \leq \log 2(\log x \vee y) \leq \log\log x + 2\log y \quad \text{for } x > e,\ y \geq 2 \tag{14}$$

that for $c_4 \geq 2$

$$\begin{aligned}&\frac{\log(T/\log(1/\delta)) \cdot \mathcal{E}_i^{-2}}{T} \cdot \log\log(T/\log(1/\delta)) \\ \leq \frac{\left[\log(c_4) + 3\log(\mathcal{E}_i^{-2})\right] \cdot \left[5\log\log(\mathcal{E}_i^{-2}) + 2\log(c_4)\right]}{c_4 \cdot \log(\mathcal{E}_i^{-2}) \cdot \log(\log(\mathcal{E}_i^{-2}) \vee e)} \\ &\leq \frac{2\log(c_4)^2 + 11\log(c_4) + 15}{c_4} \leq \frac{\log(2)}{960}\end{aligned}$$

for $c_4 \geq 7.4 \cdot 10^5$.

In the case $\mathcal{E}_i^{-2} \leq e$, we want to prove

$$\begin{aligned}\frac{(1600 + 320\log(2)^2)e}{\log(2)^3} \cdot \frac{\log(T/\log(1/\delta))}{T} \cdot \log(1/\delta) &\leq 1/3 \\ \text{for } T \geq c_2' \cdot \log(1/\delta)\ ;\end{aligned} \tag{15}$$

and

$$\begin{aligned}\frac{320e}{\log(2)} \cdot \frac{\log(T/\log(1/\delta))}{T} \cdot \log\left(\frac{1}{s_i}\right) &\leq 1/3 \\ \text{for } T \geq c_3' \cdot \log\left(\frac{1}{s_i}\right) \cdot \log\left(e \vee \log\left(\frac{1}{s_i}\right)\right)\ ;\end{aligned} \tag{16}$$

and

$$\begin{aligned}\frac{320e}{\log(2)} \cdot \frac{\log(T/\log(1/\delta))}{T} \cdot \log\log(T/\log(1/\delta)) &\leq 1/3 \\ \text{for } T \geq c_4'\ .\end{aligned} \tag{17}$$

Indeed, for inequality (15) we have

$$\frac{\log(T/\log(1/\delta))}{T} \cdot \log(1/\delta) \leq \frac{\log(c_2')}{c_2'} \leq \frac{\log(2)^3}{3e \cdot (1600 + 320\log(2)^2)}$$

for $c_2' \geq 5.7 \cdot 10^5$.

For inequality (16) we obtain

$$\frac{\log(T/\log(1/\delta))}{T} \cdot \left(\frac{1}{s_i}\right) \leq \frac{\log(c_3') + 2}{c_3'} \leq \frac{\log(2)}{960e}$$

for $c_3' \geq 4.9 \cdot 10^4$.

We finally observe that inequality (17) is true for $c_4' \geq 1.1 \cdot 10^5$, since $\delta < 1/4$.

All together, we have shown that if $T \geq c \cdot (\log(\mathcal{E}_i^{-2}) \vee 1) \cdot \log\left(\frac{\log(\mathcal{E}_i^{-2}) \vee 1}{s_i \cdot \delta}\right) \cdot \mathcal{E}_i^{-2}$ for $c \geq \max\{c_1, c_2, c_3, c_4, c_2', c_3', c_4'\}$, then $T_{d_i^*} \geq 1$ and $\Delta_i \geq 2\beta_{d_i^*}$. □

D.2 Estimation of the jumps

Algorithm 3 takes as input a set $\mathcal{I}$ of candidate intervals, a confidence level δ, a budget T, and a target number N of jump estimates. It outputs a subset $\mathcal{J} \subset \mathcal{I}$ of accepted intervals, and a set $\mathcal{G}$ of estimated jump magnitudes for the accepted intervals.

Description of the procedure. The procedure proceeds in rounds. At round k, each active interval $I \in \mathcal{I} \setminus \mathcal{J}$ receives 2^{k-1} samples at each endpoint $x_l(I)$ and $x_r(I)$ (Lines 3–4), allowing to compute empirical means $\hat{y}_l^{(k)}(I)$ and $\hat{y}_r^{(k)}(I)$. The empirical jump estimate is

$$\hat{\Delta}^{(k)}(I) = \left|\hat{y}_r^{(k)}(I) - \hat{y}_l^{(k)}(I)\right|$$

(Line 5).

An interval is accepted when this estimate exceeds a confidence threshold

$$\sqrt{2^{-(k-5)} \log\left(\frac{\pi^2 \, |\mathcal{I}| \, k^2}{3\delta}\right)}$$

(Line 6). When I is accepted at round k, its estimate $\hat{\Delta}^{(k)}(I)$ is added to $\mathcal{G}$ and the interval is added to $\mathcal{J}$ (Line 7), so it is no longer sampled in later rounds. The running budget counter τ is updated after each batch of samples (Line 9), and the while-loop (Line 2) stops once either more than N intervals are accepted ($|\mathcal{G}| \geq N$) or the next round would exceed the total budget T ($\tau + |\mathcal{I} \setminus \mathcal{J}| \cdot 2^k > T$).

This classic design yields an adaptive multiple estimation strategy: large jumps are typically accepted early, while smaller jumps receive additional rounds of sampling, under a global budget constraint. This procedure is quite standard, and is very similar to existing elimination algorithms for `TopM` identification in multi-armed bandits [see, e.g., Bubeck et al., 2013, Kalyanakrishnan et al., 2012].

Algorithm 3 `EstimateJumps`

Require: $\mathcal{I}$ set of intervals, δ confidence parameter, T budget, N number of desired estimates
Ensure: $\mathcal{J}$ subset of $\mathcal{I}$, $\mathcal{G}$ estimated jumps for intervals in $\mathcal{J}$

1: $\hat{\Delta}(I) \leftarrow 0$ for $I \in \mathcal{I}$, $k \leftarrow 1$, $\tau \leftarrow 0$, $\mathcal{G} \leftarrow \emptyset$, $\mathcal{J} \leftarrow \emptyset$
2: **while** $|\mathcal{G}| < N$ and $\tau + |\mathcal{I} \setminus \mathcal{J}| \cdot 2^k \leq T$ **do** ▷ Sample until N estimates or budget exhausted
3: **for** $[x_l(I), x_r(I)] = I \in \mathcal{I} \setminus \mathcal{J}$ **do** ▷ Sample active intervals
4: sample 2^{k-1} times from $x_l(I)$ and $x_r(I)$, store means as $\hat{y}_l^{(k)}(I)$ and $\hat{y}_r^{(k)}(I)$
5: $\hat{\Delta}^{(k)}(I) \leftarrow |\hat{y}_r^{(k)}(I) - \hat{y}_l^{(k)}(I)|$ ▷ Estimate jump at endpoints
6: **if** $\hat{\Delta}^{(k)}(I) \geq \sqrt{2^{-(k-5)} \log\left(\frac{\pi^2 \cdot |\mathcal{I}| \cdot k^2}{3\delta}\right)}$ **then** ▷ Accept if estimate exceeds threshold
7: $\mathcal{G} \leftarrow \mathcal{G} \cup \{\hat{\Delta}^{(k)}(I)\}$, $\mathcal{J} \leftarrow \mathcal{J} \cup \{I\}$ ▷ Add to accepted set
8: **end if**
9: $\tau \leftarrow \tau + 2^k$ ▷ Update budget
10: **end for**
11: $k \leftarrow k + 1$ ▷ Double samples for next round
12: **end while**

Lemma 3. *Let $M = |\mathcal{I}|$ and number the intervals $I \in \mathcal{I}$ in a way, such that if $I = [l(I), r(I)]$ and $\Delta(I) = f(r(I)) - f(l(I))$, then $|\Delta(I_1)| \geq |\Delta(I_2)| \geq \ldots |\Delta(I_M)|$. Let $\delta < 1/2$ and consider the output of Algorithm 3 $\mathcal{J} = \{J_1, \ldots, J_{N'}\}$ and $\mathcal{G} = \left\{\hat{\Delta}(J_1), \ldots, \hat{\Delta}(J_{N'})\right\}$ such that $\hat{\Delta}(J_1) \geq \cdots \geq \hat{\Delta}(J_{N'})$.*

Then there exists an event of probability at least $1 - \delta$ such that, for every $v = 1, \ldots, N'$,

$$0 < \frac{1}{2}\hat{\Delta}(J_v) \leq |\Delta(I_v)| \leq \frac{3}{2}\hat{\Delta}(J_v) \ .$$

and

$$0 < \frac{1}{2} \cdot \hat{\Delta}(J_v) \leq |\Delta(J_v)| \leq \frac{3}{2} \cdot \hat{\Delta}(J_v) \ .$$

Additionally, if

$$\begin{aligned} T \geq & c \cdot \left[\sum_{v=1}^{N} \frac{1}{\Delta(I_v)^2} \cdot \log\left(\frac{M \cdot (\log(1/\Delta(I_v)^2) \vee 1)}{\delta} \right) \right. \\ & \left. + (M - N) \cdot \frac{1}{\Delta(I_N)^2} \cdot \log\left(\frac{M \cdot (\log(1/\Delta(I_N)^2) \vee 1)}{\delta} \right) \right] \ , \end{aligned}$$

for $c \geq 223$, then on the same event $N' \geq N$.

Proof of Lemma 3. Let $k \geq 1$ be some loop iteration of the algorithm. We define the quantity $\beta_k = \sqrt{2^{-(k-3)} \log\left(\frac{\pi^2 \cdot M \cdot k^2}{3\delta}\right)}$ for twice the threshold in Line 6.

For a given interval I, the empirical means $\hat{y}_l^{(k)}(I)$ and $\hat{y}_r^{(k)}(I)$ are computed from 2^{k-1} independent samples from $f(l(I))$ and $f(r(I))$, respectively (Line 5). By Hoeffding's inequality, using the subGaussian noise assumption, for any fixed interval I and iteration k, we have

$$\mathbb{P}\left(\left| \hat{\Delta}^{(k)}(I) - \Delta(I) \right| > \beta_k \right) \leq 2 \exp\left(-\frac{2^{k-2} \cdot \beta_k^2}{2} \right) = \frac{6\delta}{\pi^2 \cdot M \cdot k^2} \ .$$

Since $\sum_{k \geq 1} \frac{1}{k^2} = \frac{\pi^2}{6}$, a union bound yields on an event ξ with $\mathbb{P}(\xi) \geq 1 - \delta$, on which

$$\left| \hat{\Delta}^{(k)}(I) - \Delta(I) \right| \leq \beta_k \tag{18}$$

for all iterations k and all intervals I considered still active at iteration k.

First, we want to show that on ξ, for every accepted interval $J_v \in \mathcal{J}$, we have $\frac{1}{2}\hat{\Delta}(J_v) \leq |\Delta(J_v)| \leq \frac{3}{2}\hat{\Delta}(J_v)$. Note that if $J_v \in \mathcal{J}$, then there exists an iteration $k^*(J_v)$ where the algorithm stops considering the interval and accepts it. In particular, at iteration $k^*(J_v)$ the condition in Line 6 is satisfied, so $\hat{\Delta}(J_v) = \hat{\Delta}^{(k^*(J_v))}(J_v)$.

By inequality (18), we know from Line 6

$$\begin{aligned} & \hat{\Delta}(J_v) \geq 2\beta_{k^*(J_v)} \\ \Rightarrow & \left| \hat{\Delta}(J_v) - |\Delta(J_v)| \right| \leq \frac{1}{2}\hat{\Delta}(J_v) \ . \end{aligned} \tag{19}$$

This proves the second inequality of Lemma 3 and if $|\Delta(J_v)| = |\Delta(I_v)|$, this coincides with the first inequality.

So assume that $|\Delta(J_v)| < |\Delta(I_v)|$. Then by inequality (19), we know that

$$\frac{1}{2}\hat{\Delta}(J_v) \leq |\Delta(J_v)| < |\Delta(I_v)| \ .$$

On the other hand, the condition $|\Delta(J_v)| < |\Delta(I_v)|$ means that there exists an $w \leq v$ such that either $I_w \notin \mathcal{J}$ or $\hat{\Delta}(I_w) \leq \hat{\Delta}(J_v)$. If $I_w \notin \mathcal{J}$ this means that the condition in Line 6 is false for any iteration, in particular at iteration $k^*(J_v)$. By the concentration inequality (18), we obtain

$$|\Delta(I_v)| \leq |\Delta(I_w)| \leq |\hat{\Delta}(I_w)^{(k^*(J_v))} - \Delta(I_w)| + \hat{\Delta}(I_w)^{(k^*(J_v))} < 3\beta_{k^*(J_v)} \leq \frac{3}{2}\hat{\Delta}(J_v) \ .$$

If $I_w \in \mathcal{J}$ but $\hat{\Delta}(I_w) \leq \hat{\Delta}(J_v)$, then there exists an iteration $k^*(I_w)$ where Algorithm 3 stops considering the interval and the condition

$$\hat{\Delta}(I_w) \geq 2\beta_{k^*(I_w)}$$

holds. If $k^*(I_w) > k^*(J_v)$, we can use the same argumentation as for the case $I_w \notin \mathcal{J}$, so assume without loss of generality that $k^*(I_w) \leq k^*(J_v)$. Then again by inequality (18) we know that

$$\begin{aligned} |\Delta(I_v)| &\leq |\Delta(I_w)| \leq |\Delta(I_w) - \hat{\Delta}(I_w)| + \hat{\Delta}(I_w) \\ &\leq \beta_{k^*(I_w)} + \hat{\Delta}(I_w) \leq \frac{3}{2}\hat{\Delta}(I_w) \leq \frac{3}{2}\hat{\Delta}(J_v) \ , \end{aligned}$$

which proves the claimed inequality.

We can prove the claim in the case $|\Delta(J_v)| > |\Delta(I_v)|$ with the same arguments. Note that in this case, there is an $w \geq v$ with $\hat{\Delta}(I_w) \geq \hat{\Delta}(J_v)$.

Now, let us prove the second part of the Lemma. Consider some interval $I \in \mathcal{I}$. Assume that $3\beta_k \leq |\Delta(I)|$. Together with inequality (18) this implies $\hat{\Delta}(I) \geq 2\beta_k$. This means, that if we reach the smallest k, such that

$$\frac{32}{9 \cdot 2^k} \log\left(\frac{\pi^2 \cdot M \cdot k^2}{3\delta}\right) \leq \Delta(I)^2 \ ,$$

before the algorithm terminates, we will have $I \in \mathcal{J}$ and I will not be considered in later sampling steps. We have assumed, that $\delta < 1/2$, so we can rewrite

$$\log\left(\frac{\pi^2 \cdot M \cdot k^2}{3\delta}\right) \leq 2\log_2(k) + 3\log\left(\frac{M}{\delta}\right) \ .$$

If we choose

$$k = \left\lceil \log_2\left(c \cdot \frac{\log\left(\frac{M \cdot (\log(1/\Delta(I)^2) \vee 1)}{\delta}\right)}{\Delta(I)^2}\right)\right\rceil$$

for some $c > 0$ large enough, we obtain with inequality (14) and $\log_2(\lceil \log_2(x) \rceil) \leq 1 + \frac{1}{\log(2)} \log(\log(x) \vee 1)$ that

$$\begin{aligned} &\frac{32}{9 \cdot \Delta(I)^2} \cdot \frac{\log\left(\frac{\pi^2 \cdot M \cdot k^2}{3\delta}\right)}{2^k} \\ &\leq \frac{32}{9} \cdot \frac{4\log\left(\frac{M}{\delta}\right) + \frac{2}{\log(2)} \cdot \log\left(\log\left(c \cdot \frac{\log\left(\frac{M \cdot (\log(1/\Delta(I)^2) \vee 1)}{\delta}\right)}{\Delta(I)^2}\right) \vee 1\right)}{c \cdot \left(\log\left(\frac{M}{\delta}\right) + \log\left(\log(1/\Delta(I)^2) \vee 1\right)\right)} \\ &\leq \cdot \frac{32(4\log(2)+4)\log\left(\frac{M}{\delta}\right) + 2\log(\log(1/\Delta(I)^2) \vee 1) + 4\log(c)}{9\log(2) \cdot c \cdot \left(\log\left(\frac{M}{\delta}\right) + \log\left(\log(1/\Delta(I)^2) \vee 1\right)\right)} \\ &\leq \frac{128(\log(2)+1)}{9\log(2)} \frac{1 + \log(c)}{c} \leq 1 \end{aligned}$$

for $c \geq 223$.

So for sampling the arms corresponding to interval I, we use a budget of at most

$$4c \cdot \frac{\log\left(\frac{M \cdot (\log(1/\Delta(I)^2) \vee 1)}{\delta}\right)}{\Delta(I)^2}$$

and spend the remaining budget on the intervals in $\mathcal{I} \setminus \mathcal{J}$. The algorithm stops, when we added N arms to $\mathcal{J}$, which yields the claimed bound of the total budget. □

D.3 Single change point localization

Once we have detected an interval I containing a change point, we want to localize it, at a precision of η. We adapt for that the `Sequential halving with backtracking` (`SHB`) algorithm introduced in Lazzaro and Pike-Burke [2025b] as their Algorithm 2, and recall their optimal guarantees for single change point localization, with fixed budget.

Description of the procedure. The algorithm takes as input a compact interval $I = [l(I), r(I)]$, a budget T, and a precision parameter $\eta > 0$. It outputs a potential change point c.

The algorithm maintains a tuple of five point arms $(l(I), l_d, c_d, r_d, r(I))$, where the boundary arms $l(I)$ and $r(I)$ are fixed throughout. At each round d, it collects τ samples from each arm and compares the resulting empirical means to decide whether to zoom into the left sub-interval (l_d, c_d), the right sub-interval (c_d, r_d). It may also *backtrack* to the parent window when the evidence suggests the change point lies outside the current candidate region [Lazzaro and Pike-Burke, 2025b]. The procedure runs for $d_{\max} = \lceil 6 \log(|I|/\eta) \rceil$ rounds, distributing the budget evenly across rounds and arms.

The only (and minor) difference with the original `SHB` algorithm is that we stop the procedure early if $|I| \leq 2\eta$, and return the midpoint of the interval as the change point estimate. This is because in this case, we are already guaranteed to be within η of the true change point, so there is no need to further zoom in. Moreover, we adapt the procedure to an interval of size $|I|$ instead of $[0, 1]$, which only requires a rescaling of the arms and the precision parameter η.

Algorithm 4 `SHB (Sequential Halving with Backtracking)`

Require: $I = [l(I), r(I)]$ compact interval, T budget, $\eta > 0$ precision parameter
Ensure: potential change point c

$c_1 \leftarrow \frac{l(I)+r(I)}{2}$, $l_1 \leftarrow l(I)$, $r_1 \leftarrow r(I)$
if $|I| \leq 2\eta$ **then**
  return c_1 ▷ Return interval midpoint if target precision already achieved
else
  $d_{\max} \leftarrow \lceil 6 \log(|I|/\eta) \rceil$, $\tau \leftarrow \lfloor T/5d_{\max} \rfloor$, $\mathcal{A}^1 \leftarrow (l(I), l_1, c_1, r_1, r(I))$
  for $d = 1, \ldots, d_{\max}$ **do**
    sample τ times from arms in $\mathcal{A}^d$, store means as $\hat{y}^{(d)}_{l(I)}$, $\hat{y}^{(d)}_{l_d}$, $\hat{y}^{(d)}_{c_d}$, $\hat{y}^{(d)}_{r_d}$, $\hat{y}^{(d)}_{r(I)}$
    if $\left| \frac{\hat{y}^{(d)}_{l(I)} + \hat{y}^{(d)}_{l_d}}{2} - \frac{\hat{y}^{(d)}_{r_d} + \hat{y}^{(d)}_{r(I)}}{2} \right| < \frac{3}{4} \max \left(\left| \frac{\hat{y}^{(d)}_{l(I)} + \hat{y}^{(d)}_{l_d} + \hat{y}^{(d)}_{r_d}}{3} - \hat{y}^{(d)}_{r(I)} \right|, \left| \hat{y}^{(d)}_{l(I)} - \frac{\hat{y}^{(d)}_{l_d} + \hat{y}^{(d)}_{r_d} + \hat{y}^{(d)}_{r(I)}}{3} \right| \right)$ **then** ▷ If Jump seems outside middle region
      $\mathcal{A}^{d+1} \leftarrow P(\mathcal{A}^d)$ ▷ Backtrack: retreat to parent region
    else if $|\hat{y}^{(d)}_{l_d} - \hat{y}^{(d)}_{c_d}| \leq |\hat{y}^{(d)}_{c_d} - \hat{y}^{(d)}_{r_d}|$ **then** ▷ Jump seems closer to right
      $\mathcal{A}^{d+1} \leftarrow R(\mathcal{A}^d) = (l(I), c_d, (c_d + r_d)/2, r_d, r(I))$ ▷ Zoom right
    else ▷ Jump seems closer to left
      $\mathcal{A}^{d+1} \leftarrow L(\mathcal{A}^d) = (l(I), l_d, (l_d + c_d)/2, c_d, r(I))$ ▷ Zoom left
    end if
  end for
  return c_{d+1} ▷ Return final midpoint
end if

Lemma 4 (Lazzaro and Pike-Burke, 2025b)**.** *Assume that $x^*_{i-1} \leq l(I) < x^*_i \leq r(I) < x^*_{i+1}$. If $|I| \leq 2\eta$, Algorithm 4 will return $c = \frac{l(I)+r(I)}{2}$ with $|x^*_i - c| \leq \eta$.*

Else, for a budget

$$T \geq \frac{600}{\Delta_i^2} \left(\log\left(\frac{1}{\delta}\right) + 13 \log\left(\frac{|I|}{4\eta}\right) \right)$$

Algorithm 4 will return c with $|x^*_i - c| \leq \eta$ with probability at least $1 - \delta$.

Proof. The first part is immediate: if $|I| \leq 2\eta$, the midpoint $c = \frac{l(I)+r(I)}{2}$ satisfies $|c - x^*_i| \leq |I|/2 \leq \eta$ for any $x^*_i \in [l(I), r(I)]$. For the second part, we apply Theorem 1 from Lazzaro and

Pike-Burke [2025b] directly.[8]). Our model is right-continuous, so Lemma 4 is its mirror image; the theorem applies after a left-right reflection. □

D.4 Verification of the localized change points

Description of the procedure. `VerifyCP` is given in Algorithm 5, and is a very standard two-sample test. It takes as input a candidate change point location x, an interval I containing x, a confidence parameter δ, a precision parameter η, and a budget T. It performs a simple two-sample test to check for the presence of a change point in the neighborhood of x, given by the interval $[x_-, x_+]$ where $x_- = (x - \eta) \vee \min I$ and $x_+ = (x + \eta) \wedge \max I$, which corresponds to an η-neighborhood around x included in I. As we see in Lemma 5, if there is no actual jump ($f(x_+) = f(x_-)$), the algorithm correctly returns `False` with high probability $1 - \delta$. Conversely, if there is a jump of size $\Delta = |f(x_+) - f(x_-)|$, and the budget T is sufficiently large (scaling as $1/\Delta^2$), the algorithm will correctly return `True` with high probability.

Algorithm 5 `VerifyCP`

Require: x_- left point, x_+ right point, δ confidence parameter, T budget
Ensure: `detection` (boolean)
  sample $\lfloor T/2 \rfloor$ times from x_- and x_+, store averages as $\hat{y}_-$ and $\hat{y}_+$
  if $T \geq 2$ and $|\hat{y}_+ - \hat{y}_-| > \sqrt{\frac{16}{T}\log(2/\delta)}$ **then**
    `detection` $\leftarrow$ `True`
  else
    `detection` $\leftarrow$ `False`
  end if

Lemma 5. *Let $\delta < 1/2$ and consider the output of Algorithm 5 with input x_-, x_+, δ, and T. Let $\Delta := f(x_+) - f(x_-)$.*

1. *If $\Delta = 0$, with probability at least $1 - \delta$ Algorithm 5 returns `False`.*
2. *If $\Delta \neq 0$ and $T \geq 64 \cdot \frac{\log\left(\frac{2}{\delta}\right)}{\Delta^2}$, Algorithm 5 returns `True` with probability at least $1 - \delta$.*

Proof of Lemma 5. By Hoeffding's inequality, we know that for

$$\beta = \sqrt{\frac{16}{T}\log(2/\delta)}$$

holds

$$\mathbb{P}\left(|(\hat{y}_+ - \hat{y}_-) - (y_+ - y_-)| \geq \beta\right) \leq \delta \ .$$

So if $y_+ = y_-$, then $|\hat{y}_+ - \hat{y}_-| < \beta$ and Algorithm 5 returns `False`, with probability at least $1 - \delta$. If on the other hand $\Delta = |y_+ - y_-| \geq 2\beta$ this guarantees, that $|\hat{y}_+ - \hat{y}_-| > \beta$ and Algorithm 5 will return `True`. This yields the condition

$$\Delta \geq 2 \cdot \sqrt{\frac{16}{T}\log(2/\delta)} \quad \Leftrightarrow \quad T \geq 64 \cdot \frac{\log(2/\delta)}{\Delta^2} \ . \qquad \square$$

D.5 Proof of the main theorem

Proof of Theorem 1. In this proof, we gather the guarantees on our four subroutines through the following Lemmas: (i) Lemma 1 for `DetectIntervals`, (ii) Lemma 3 for `EstimateJumps`, (iii) Lemma 4 for `SHB`, (iv) Lemma 5 for `VerifyCP`.

First, we justify the correctness guarantee in point 1 of the theorem. Then, we fix a stage index k in the doubling schedule and analyze the four steps of the procedure at this stage, deriving conditions on k under which they provide an accurate estimate of N change points and the stopping condition is

[8] Lazzaro and Pike-Burke [2025b] uses a left-endpoint convention (f left-continuous, change point admissible at the left endpoint of the interval

reached. Finally, we derive a high-probability bound on the required budget when $\delta_{\text{explore}} = \delta$, and an expectation bound when $\delta_{\text{explore}} = 1/4$.

Consider δ, η and N as in the statement of the theorem, and any environment ν with $m \geq N$ change points. We run Algorithm 2 with input parameters $(N, \delta, \eta, \delta_{\text{explore}})$, with $\delta_{\text{explore}} \in \{\delta, 1/4\}$.

Correction. The correctness guarantee relies only on the verification step and is independent of the choice of δ_{explore}. Let us verify that if Algorithm 2 stops at some stage k, the set $\mathcal{C}^{(k)}$ returned by the algorithm contains N points $c_1^{(k)} < \cdots < c_N^{(k)}$ with $|x^*_{l_v} - c_v^{(k)}| \leq \eta$ for N distinct change points $1 \leq l_1 < l_2 < \cdots < l_N \leq m$.

If not, then there exists a stage index k, and some estimated change point $c_v^{(k)} \in J_v^{(k)}$, with $J_v^{(k)} \in \mathcal{J}^{(k)}$, such that $[c_v^{(k)} - \eta, c_v^{(k)} + \eta] \cap J_v^{(k)}$ does not contain any change point, but $\text{ok}_v^{(k)} = \texttt{VerifyCP}\left(l_v^{(k)}, r_v^{(k)}, \delta^{(k)}, \left\lfloor \alpha_v^{(k)} \cdot 2^k \right\rfloor \vee 1\right)$ is True.

Consider such a $c_v^{(k)}$. Then, in `VerifyCP`, for $l_v^{(k)} = (c_v^{(k)} - \eta) \vee \min(J_v^{(k)})$ and $r_v^{(k)} = (c_v^{(k)} + \eta) \wedge \max(J_v^{(k)})$, we have $f(l_v^{(k)}) = f(r_v^{(k)})$. By Lemma 5, there exists an event $\xi^{(k)}_{\text{verify},v}$ with probability at least $1 - \delta^{(k)}$ such that `VerifyCP` in Line 10 returns `False` for $c_v^{(k)}$. By a union bound, the probability of returning an incorrect set $C^{(k)}$ in step k is therefore bounded by $N\delta^{(k)} = \frac{\delta}{4} \cdot \frac{6}{\pi^2 (k)^2}$. Now, since

$$\sum_{k'=\lceil \log_2(N) \rceil}^{+\infty} \frac{\delta}{4} \cdot \frac{6}{\pi^2 (k')^2} \leq \frac{1}{4}\delta \ ,$$

another union bound tells us that, with probability at least $1 - \delta$, the set $\mathcal{C}$ returned by Algorithm 2 contains estimates of N distinct change points at uniform precision η. This proves the correctness guarantee in point 1 of Theorem 1.

Construction of a good event for a fixed stage index k. Fix for now a stage index k in the doubling schedule (the index of the global while loop), at which the budget used is $T_k = 4 \times 2^k$. We analyze the four steps of the procedure at this stage, and we derive conditions on k under which they provide an accurate estimate of N change points and the stopping condition is reached with probability at least $1 - \delta_{\text{explore}}$.

(i) Detection of changes in intervals of interest.

For a fixed stage index k, consider the set $\mathcal{I}^{(k)}$ of intervals returned by `DetectIntervals` in Line 3 of Algorithm 2, with confidence parameter $\delta_{\text{explore}}/4$, and budget $T_k/4 = 2^k$.

Consider $\omega_i^a(\delta_{\text{explore}}/4)$, and $c_1 = 7.4 \cdot 10^5$ numerical constant from Lemma 1. By Lemma 1, under the condition

$$2^k \geq c_1 \cdot \max_{i=1,\ldots,N} \omega_i^a \left(\frac{\delta_{\text{explore}}}{4}\right) \cdot \mathcal{E}_i^{-2} \ , \tag{20}$$

there is an event $\xi^{(k)}_{\text{detect}}$ with $\mathbb{P}\left(\xi^{(k)}_{\text{detect}}\right) \geq 1 - \delta_{\text{explore}}/4$, on which the set $\mathcal{I}^{(k)}$ consists of m disjoint intervals, each containing exactly one change point. Denote by $I_i^{(k)}$ the interval in $\mathcal{I}^{(k)}$ with $x_i^* \in I_i^{(k)}$; it follows from the lemma that $|I_i^{(k)}| \leq \frac{s_i}{2}$.

Note that since $\delta_{\text{explore}} < 1/4$ we have

$$\max_{i=1,\ldots,N} \omega_i^a \left(\frac{\delta_{\text{explore}}}{4}\right) \cdot \mathcal{E}_i^{-2} \leq 2\left(\omega_1 H^{(m)}_{\text{detect}} \left(\log\left(\frac{1}{\delta_{\text{explore}}}\right) + \omega_2\right)\right) \ ,$$

so that the condition (20) is satisfied as soon as

$$2^k \geq 2c_1 \cdot \omega_1 H^{(m)}_{\text{detect}} \left(\log\left(\frac{1}{\delta_{\text{explore}}}\right) + \omega_2\right) \ . \tag{21}$$

Assume for what follows that the event $\xi^{(k)}_{\text{detect}}$ holds, and that condition (20) is satisfied.

(ii) Estimation of the jumps.

Consider the subset $\mathcal{J}^{(k)} \subset \mathcal{I}^{(k)}$ of intervals returned by `EstimateJumps` in Line 5 of Algorithm 2, with confidence parameter $\delta_{\text{explore}}/4$ and budget $T_k/4 = 2^k$, and number of arms N. Let $\mathcal{G}^{(k)}$ denote the set of estimated jump magnitudes on the intervals in $\mathcal{J}^{(k)} = \{J_v^{(k)}\}_{v=1}^{N'}$, ordered decreasingly, so that $\hat{\Delta}(J_1^{(k)}) \geq \cdots \geq \hat{\Delta}(J_{N'}^{(k)})$, with $N' = |\mathcal{J}^{(k)}|$. On $\xi_{\text{detect}}^{(k)}$, the intervals in $\mathcal{I}^{(k)}$ are disjoint and each contains exactly one change point. In particular, the true jumps on these intervals are exactly the jump magnitudes of the m change points of f.[9] Therefore, we can apply Lemma 3 to the set of intervals $\mathcal{I}^{(k)}$ returned by `DetectIntervals` and to the corresponding jumps $\Delta_1, \ldots, \Delta_m$.

By Lemma 3, there exists an event $\xi_{\text{estimate}}^{(k)}$ with $\mathbb{P}(\xi_{\text{estimate}}^{(k)}) \geq 1 - \delta_{\text{explore}}/4$, so that, with c_2 a numerical constant from Lemma 3, under the condition

$$2^k \geq c_2 \cdot \left[\sum_{i=1}^{N} \frac{1}{\Delta_{(i)}^2} \cdot \log\left(\frac{4m \cdot (\log(1/\Delta_{(i)}^2) \vee 1)}{\delta_{\text{explore}}} \right) + (m-N) \cdot \frac{1}{\Delta_{(N)}^2} \cdot \log\left(\frac{4m \cdot (\log(1/\Delta^2(I_{(N)})) \vee 1)}{\delta_{\text{explore}}} \right) \right] \tag{22}$$

`EstimateJumps` returns a set $\mathcal{G}^{(k)} = \{\hat{\Delta}(J_v^{(k)}),\ v = 1, \ldots, N'\}$, with $N' \geq N$, with $\hat{\Delta}(J_1^{(k)}) \geq \hat{\Delta}(J_2^{(k)}) \geq \cdots \geq \hat{\Delta}(J_{N'}^{(k)})$, and such that for $v = 1, \ldots, N$,

$$\frac{1}{2}\hat{\Delta}(J_v^{(k)}) \leq |\Delta(J_v^{(k)})| \leq \frac{3}{2}\hat{\Delta}(J_v^{(k)}) \tag{23}$$

$$\frac{1}{2}\hat{\Delta}(J_v^{(k)}) \leq |\Delta_{(v)}| \leq \frac{3}{2}\hat{\Delta}(J_v^{(k)}) \ . \tag{24}$$

These inequalities will allow us to verify that the proportion $\alpha_v^{(k)}$ of the budget allocated to interval $J_v^{(k)}$ is proportional to $\Delta_{(v)}^{-2}$, which we interpret as the optimal allocation for localizing the change point in $J_v^{(k)}$.

Let us now examine condition (22). Naturally, estimating the jumps once they have been detected is less costly than the detection step itself. First, the energy $\mathcal{E}_i^2$ is the product of Δ_i^2 and a spacing term, so $\Delta_i^{-2} \leq \mathcal{E}_i^{-2}$ for all i. This implies that all logarithmic terms in the condition (22) are upper bounded by $2(\omega_2 + \log(1/(\delta_{\text{explore}})))$, with ω_2 defined in the statement of the theorem. Secondly, one has

$$\sum_{i=1}^{N} \frac{1}{\Delta_{(i)}^2} + (m-N) \cdot \frac{1}{\Delta_{(N)}^2} \leq \sum_{i=1}^{m} \frac{1}{\Delta_{(i)}^2} \leq \sum_{i=1}^{m} \frac{1}{s_i \Delta_{(i)}^2} s_i \leq \left(\max_{j=1,\ldots,m} \mathcal{E}_j^{-2} \right) \sum_{i=1}^{m} s_i \leq 2H_{\text{detect}}^{(m)} \ ,$$

where the last inequality holds since $\sum_{i=1}^{m} s_i \leq 2$, and $H_{\text{detect}}^{(m)} = \max_{i=1,\ldots,m} \mathcal{E}_i^{-2}$.

All together, condition (22) is implied by

$$2^k \geq 2c_2 \cdot H_{\text{detect}}^{(m)} \cdot (\omega_2 + \log(1/\delta_{\text{explore}})) \tag{25}$$

Now, we assume that both events $\xi_{\text{detect}}^{(k)}$ and $\xi_{\text{estimate}}^{(k)}$ hold, and that conditions (20) and (22) are satisfied. We are now in a position to analyze the last two steps of the procedure.

(iii) Localization of the change points.

Let $v \in \{1, \ldots, N\}$, and consider the interval $J_v^{(k)}$ returned by `EstimateJumps`, with the v-th largest estimated jump $\hat{\Delta}(J_v^{(k)})$. From the guarantees on `DetectIntervals`, we know that $J_v^{(k)}$ contains exactly one change point, with jump $\Delta(J_v^{(k)})$, and that the if this unique change point is $x_{j_v}^*$, then $|\Delta(J_v^{(k)})| = |\Delta_{j_v}|$, and $|J_v^{(k)}| \leq s_{j_v}$. Moreover, by Lemma 1 and its proof, $x_{j_v-1}^* < l(J_v^{(k)}) < x_{j_v}^* \leq r(J_v^{(k)}) < x_{j_v+1}^*$, so that the assumption of Lemma 4 is satisfied.

[9] Recall that we defined $\Delta(I) = f(l) - f(r)$ for the jump across an interval $I = [l, r]$.

Consider the proportion $\alpha_v^{(k)}$ of the budget allocated to interval $J_v^{(k)}$ for localization, defined in Line 9 of Algorithm 2. Let $c_v^{(k)}$ denote the output of the SHB algorithm (Algorithm 4) applied to $J_v^{(k)}$ with budget $\lfloor \alpha_v^{(k)} \cdot 2^k \rfloor \vee 1$, where $\alpha_v^{(k)}$ is defined in (6).

If $|J_v^{(k)}| \leq 2\eta$, Algorithm 4 takes the midpoint of the interval as the change point estimate $c_v^{(k)}$, and we have the guarantee $|c_v^{(k)} - x_{j_v}^*| \leq \eta$. Otherwise, by Lemma 4 from Lazzaro and Pike-Burke [2025b], there exists an event $\xi_{\mathrm{loc},v}^{(k)}$ of probability at least $1 - \delta_{\mathrm{explore}}/4N$ such that, under the condition

$$\lfloor \alpha_v^{(k)} \cdot 2^k \rfloor \vee 1 \geq \frac{7200}{\Delta_{j_v}^2} \log\left(\frac{N}{\eta \cdot \delta_{\mathrm{explore}}}\right) \quad , \tag{26}$$

an estimate $c_v^{(k)} \in J_v^{(k)}$ with $|c_v^{(k)} - x_{j_v}^*| \leq \eta$.

Let us now find a condition on k that implies the condition (26) for any $v = 1, \ldots, N$. From Equations (23) and (24), we have $|\Delta(J_v^{(k)})| \geq \frac{1}{2}\hat{\Delta}(J_v^{(k)}) \geq \frac{1}{2}\frac{2}{3}\Delta_{(v)} = \frac{1}{3}\Delta_{(v)}$, where $\Delta_{(v)}$ is the v-th largest true jump. In particular, this implies that

$$H_{\mathrm{localize}}^{(N)} = \sum_{w=1}^{N} \Delta_{(w)}^{-2} \geq \frac{1}{9} \sum_{w=1}^{N} \Delta(J_w^{(k)})^{-2} \quad .$$

Moreover, from inequality (23), we also obtain for any $v = 1, \ldots, N$,

$$\alpha_v^{(k)} = \frac{\left(\hat{\Delta}(J_v)\right)^{-2}}{\sum_{w=1}^{N}\left(\hat{\Delta}(J_w)\right)^{-2}} \geq \frac{(2\Delta_{j_v})^{-2}}{\sum_{w=1}^{N}\left(\frac{2}{3}\Delta_{j_w}\right)^{-2}} = \frac{1}{9}\frac{\Delta_{j_v}^{-2}}{\sum_{w=1}^{N}\Delta_{j_w}^{-2}} \quad .$$

Now, under the assumption

$$2^k \geq 1263600 \log\left(\frac{N}{\eta \cdot \delta_{\mathrm{explore}}}\right) \sum_{w=1}^{N} \frac{1}{\Delta_{(w)}^2} \geq 140400 \log\left(\frac{N}{\eta \cdot \delta_{\mathrm{explore}}}\right) \sum_{w=1}^{N} \frac{1}{\Delta_{j_w}^2}$$

we obtain

$$\left\lfloor \alpha_{i_v}^{(k)} \cdot 2^k \right\rfloor \geq \frac{7200}{\Delta_{j_v}^2} \log\left(\frac{N}{\eta \cdot \delta_{\mathrm{explore}}}\right) \quad .$$

By a union bound, we obtain an event $\xi_{\mathrm{localize}}^{(k)} = \cap_{v=1}^{N} \xi_{\mathrm{loc},v}^{(k)}$ with $\mathbb{P}(\xi_{\mathrm{localize}}^{(k)}) \geq 1 - \delta/4$, on which $|c_v^{(k)} - x_{j_v}^*| \leq \eta$. This yields following sufficient condition for localization:

$$2^k \geq 1263600 \cdot H_{\mathrm{localize}}^{(N)} \cdot \left(\log\left(\frac{1}{\delta_{\mathrm{explore}}\eta}\right) + \omega_3\right) \quad , \tag{27}$$

using $\omega_3 = \log\left(N \cdot \log(H_{\mathrm{localize}}^{(N)})\right)$ as defined in the theorem.

For now, assume that we are on the event $\xi_{\mathrm{detect}}^{(k)} \cap \xi_{\mathrm{estimate}}^{(k)} \cap \xi_{\mathrm{localize}}^{(k)}$.

(iv) Verification of the localized change points.

Finally, we seek a condition on k such that, if the previous steps go well, the verification step also succeeds and the algorithm stops. This will hold under the following condition:

$$2^k = c' \cdot \log\left(\frac{N \cdot \left(\log\left(\sum_{w=1}^{N} \frac{1}{\Delta_{(w)}^2}\right) \vee 1\right)}{\delta}\right) \cdot \sum_{w=1}^{N} \frac{1}{\Delta_{(w)}^2},$$

where $c' > 0$ is computed later. If we can show that

$$\alpha_{i_v}^{(k)} \cdot 2^k \geq 128 \cdot \frac{\log\left(\frac{4\pi^2 \cdot N \cdot k^2}{3 \cdot \delta}\right)}{\Delta_{j_v}^2} \quad , \quad i = 1, \ldots, N \quad ,$$

then

$$\left\lfloor \alpha_{i_v}^{(k)} \cdot 2^k \right\rfloor \geq 64 \cdot \frac{\log\left(\frac{4\pi^2 \cdot N \cdot k^2}{3 \cdot \delta}\right)}{\Delta_{j_v}^2} \quad , \quad v = 1, \ldots, N \ ,$$

and Lemma 5 together with a union bound show that, on the event $\xi_{\text{verify}}^{(k)}$ with $\mathbb{P}(\xi_{\text{verify}}^{(k)}) \geq 1 - \frac{3\delta}{2\pi^2 k^2}$ (already introduced in the correctness proof), `VerifyCP` in Line 10 returns `True` for all $i = 1, \ldots, N$.

Note that we have by the inequalities (23) and (24) that

$$\begin{aligned}
&\alpha_{i_v}^{(k)} \cdot c' \cdot \log\left(\frac{N \cdot \left(\log\left(\sum_{w=1}^N \frac{1}{\Delta_{(w)}^2}\right) \vee 1\right)}{\delta}\right) \cdot \sum_{w=1}^N \frac{1}{\Delta_{(w)}^2} \\
&\geq \frac{c'}{9\Delta_{j_v}^2} \cdot \log\left(\frac{N \cdot \left(\log\left(\frac{1}{9}\sum_{i=1}^N \frac{1}{\Delta_{j_w}^2}\right) \vee 1\right)}{\delta}\right) \\
&\geq \frac{c'}{18\Delta_{j_v}^2} \cdot \log\left(\frac{N \cdot \left(\log\left(\sum_{i=1}^N \frac{1}{\Delta_{j_w}^2}\right) \vee 1\right)}{\delta}\right) \quad .
\end{aligned}$$

The last inequality follows from the fact that for $\delta < 1/4$ and $x > 0$, we can bound

$$\log\left(\frac{N \cdot (\log(\frac{1}{9}x) \vee 1))}{\delta}\right) \geq \frac{1}{2}\log\left(\frac{N \cdot (\log(x) \vee 1)}{\delta}\right) \ ,$$

since

$$\begin{aligned}
&\log\left(\frac{N \cdot (\log(\frac{1}{9}x) \vee 1))}{\delta}\right) - \frac{1}{2}\log\left(\frac{N \cdot (\log(x) \vee 1)}{\delta}\right) \\
&= \frac{1}{2}\log\left(\frac{N}{\delta}\right) + \log\left(\log\left(\frac{1}{9}x\right) \vee 1\right) - \frac{1}{2}\log\left(\log\left(x\right) \vee 1\right) \\
&\geq \log(2) + \log\left(\left(\log\left(x\right) - \log(9)\right) \vee 1\right) - \frac{1}{2}\log\left(\log\left(x\right) \vee 1\right) \quad =: g(x).
\end{aligned}$$

The function g is continuous $[0, \infty)$ with $g(x) = \log(2)$ on $[0, e]$, $g(x)$ decreasing on $(e, 9e)$ with $g(9e) = \log(2) - \frac{1}{2}\log(1 + \log(9)) \geq 0$ and increasing on $(9e, \infty)$, so we have indeed $g(x) \geq 0$.

Therefore, we obtain

$$\begin{aligned}
\frac{\log\left(\frac{4\pi^2 \cdot N \cdot k^2}{3 \cdot \delta}\right)}{\alpha_{i_v}^{(k)} \cdot 2^k \cdot \Delta_{j_v}^2} &\leq \frac{182\log\left(\frac{N}{\delta}\right) + 102\log\left(\log\left(\sum_{w=1}^N \frac{1}{\Delta_{(w)}^2}\right) \vee 1\right) + 72\log(c')}{c' \cdot \left[\log\left(\log\left(\sum_{w=1}^N \frac{1}{\Delta_{j_w}^2}\right) \vee 1\right) + \log\left(\frac{N}{\delta}\right)\right]} \\
&\leq \frac{182 + 72\log(c_4)}{c_4} \leq \frac{1}{128}
\end{aligned}$$

when $c_4 > 1.4 \cdot 10^5$. For the second inequality, we used Equation (14) to obtain

$$\begin{aligned}
\log\log(2^k) &\leq \log\left(\log\left(\sum_{w=1}^N \frac{1}{\Delta_{(w)}^2}\right) \vee 1\right) \\
&\quad + 2\log\left(c' \cdot \log\left(\frac{N \cdot \left(\log\left(\sum_{w=1}^N \frac{1}{\Delta_{(w)}^2}\right) \vee 1\right)}{\delta}\right)\right) \\
&\leq 2\log\left(\frac{N}{\delta}\right) + 3\log\left(\log\left(\sum_{w=1}^N \frac{1}{\Delta_{(w)}^2}\right) \vee 1\right) + 2\log(c') \ .
\end{aligned}$$

All together, using $\omega_3 = \log\left(N \cdot \left(\log\left(H^{(N)}_{\text{localize}}\right) \vee 1\right)\right)$, we have shown that under the condition

$$2^k \geq 1.4 \cdot 10^5 \cdot (\omega_3 + \log(1/\delta)) \cdot H^{(N)}_{\text{localize}} \ , \tag{28}$$

then on the event $\xi^{(k)}_{\text{verify}}$, `VerifyCP` in Line 10 returns `True` for all $v = 1, \ldots, N$.

Bound in high probability. Now, we are ready to derive a high probability bound on the required budget when $\delta_{\text{explore}} = \delta$. We have a constant c such that for any $k \geq \lceil \log_2(N) \rceil$, if k satisfies

$$2^k \geq c \cdot \left(\omega_1 H^{(m)}_{\text{detect}} \left(\log\left(\frac{1}{\delta}\right) + \omega_2\right) + H^{(N)}_{\text{localize}} \left(\log\left(\frac{1}{\delta\eta}\right) + \omega_3\right)\right) \ , \tag{29}$$

then all conditions (21),(25),(27),(28) are also verified (with $\delta_{\text{explore}} = \delta$). From the previous paragraph, under this budget, on the intersection $\xi^{(k)} := \xi^{(k)}_{\text{detect}} \cap \xi^{(k)}_{\text{estimate}} \cap \xi^{(k)}_{\text{localize}} \cap \xi^{(k)}_{\text{verify}}$, the stopping condition at the k-th stage of Algorithm 2 is reached. It is in particular sufficient to choose k_* as the smallest integer such that this bound (29) holds. Now, on the event $\xi = \xi^{(k_*)}_{\text{detect}} \cap \xi^{(k_*)}_{\text{estimate}} \cap \xi^{(k_*)}_{\text{localize}} \cap (\cap_{k'=1}^{k_*} \xi^{(k')}_{\text{verify}})$, then the algorithm stops at stage k_* (or before), and the output is correct.

Finally, one has $\mathbb{P}(\xi^{(k_*)}) \geq 1 - \frac{3}{4}\delta - \sum_{k'=1}^{k_*} \frac{6}{\pi^2 k'^2} \cdot \frac{\delta}{4} \geq 1 - \delta$. Moreover, on ξ, the total budget used by the algorithm can be bounded as $\sum_{k'=\lceil \log_2(N) \rceil}^{k_*} 4 \cdot 2^{k'} < 8 \cdot 2^{k_*}$, which gives us the claimed bound on the total budget. This concludes the proof of the high probability bound in point 2 of Theorem 1.

Bound in expectation. Finally, we can also derive a bound on the expected budget. Fix $\delta_{\text{explore}} = 1/4$.

From the analysis of the four steps of the procedure, there exists a universal constant c such that for any $k \geq \lceil \log_2(N) \rceil$, if k satisfies

$$2^k \geq c \cdot \left(\omega_1 H^{(m)}_{\text{detect}} \left(\log(4) + \omega_2\right) + H^{(N)}_{\text{localize}} \left(\log(4/\delta\eta) + \omega_3\right)\right) \ , \tag{30}$$

then all conditions (21),(25),(27),(28) are verified. Under this budget constraint, on the intersection $\xi^{(k)} := \xi^{(k)}_{\text{detect}} \cap \xi^{(k)}_{\text{estimate}} \cap \xi^{(k)}_{\text{localize}} \cap \xi^{(k)}_{\text{verify}}$, the stopping condition at the k-th stage of Algorithm 2 is reached. Moreover, $\mathbb{P}(\xi^{(k)}) \geq 1 - \frac{3}{4} \cdot \frac{1}{4} - \sum_{k'=\lceil \log_2(N) \rceil}^{k} \frac{6}{\pi^2 k'^2} \cdot \frac{\delta}{4} \geq \frac{3}{4}$, as $\delta \leq 1/4$ and $\delta_{\text{explore}} = 1/4$.

Now, consider k_0, the smallest integer such that condition (30) is satisfied. Then, on the event $\xi^{(k_0)}$, for any $k \geq k_0$, the stopping condition at the k-th stage of Algorithm 2 is reached, and the output is correct with probability at least $\frac{3}{4}$. Therefore, if we denote by K the random variable corresponding to the stage at which the algorithm stops, then $(K - k_0) \vee 0$ is stochastically dominated by a geometric random variable with parameter $3/4$, and $\mathbb{E}[2^{(K-k_0)\vee 0}] \leq \sum_{l=0}^{\infty} 2^l (1/4)^l (3/4) = \frac{3}{2}$.

Moreover, the total budget used by the algorithm can be bounded as

$$\mathcal{T} \leq \sum_{k'=\lceil \log_2(N) \rceil}^{K} 4 \cdot 2^{k'} < 8 \cdot 2^{k_0} \cdot 2^{(K-k_0)\vee 0} \ ,$$

and

$$\begin{aligned} \mathbb{E}[\mathcal{T}] &\leq 8 \cdot 2^{k_0} \cdot \mathbb{E}[2^{(K-k_0)\vee 0}] \leq 12 \cdot 2^{k_0} \\ 2^{k_0} &\leq 2c \cdot \left(\omega_1 H^{(m)}_{\text{detect}} \left(\log(4) + \omega_2\right) + H^{(N)}_{\text{localize}} \left(\log(4/\delta\eta) + \omega_3\right)\right) \ . \end{aligned}$$

This concludes the proof of the bound in expectation in point 3 of Theorem 1. □

E Proofs of lower bounds

E.1 Proof of Theorem 2 and Corollary 1

Before proving Theorem 2, we first state two lemmas, which are the main building blocks of the proof. The first is a lower bound showing the optimality of the localization complexity $H^{(m)}_{\mathrm{localize}}$, and the second is a lower bound scaling with $H^{(m)}_{\mathrm{detect}}$.

Fix $\delta \in (0, 1/4)$ and $\eta \in (0, 1/8)$. For both lemmas, we consider any (δ, η, m)-correct algorithm π for the active change point detection problem, and any valid environment ν with $m \geq 1$ change points. We consider the case where all change points have to be localized, so that $N = m$. In all this section, we may omit the superscript (m) in $H^{(m)}_{\mathrm{detect}}$ and $H^{(m)}_{\mathrm{localize}}$ for readability, and simply write H_{detect} and H_{localize}. We may precise the dependence of these quantities on the environment ν when needed, and write $H^{\nu}_{\mathrm{detect}}$ and $H^{\nu}_{\mathrm{localize}}$.

Lemma 6. *There exists an environment ν' such that (1) ν' and ν have the same jump magnitudes $\Delta_1, \dots, \Delta_m$ – so that $H^{(m,\nu')}_{\mathrm{localize}} = H^{(m,\nu)}_{\mathrm{localize}}$; (2) $\forall i \in \{1, \dots, m\}$, $\frac{1}{4}s_i \leq s'_i \leq \frac{3}{4}s_i$ – so that $\frac{4}{3}H^{(m,\nu)}_{\mathrm{detect}} \leq H^{(m,\nu')}_{\mathrm{detect}} \leq 4H^{(m,\nu)}_{\mathrm{detect}}$; and*

$$\mathbb{P}_{\pi,\nu'}\left(\mathcal{T}_\pi \geq H^{(m,\nu)}_{\mathrm{localize}} \log\left(\frac{1}{8\delta}\right) + \sum_{i=1}^{m} \frac{1}{\Delta_i^2} \log_+\left(\frac{s_i}{16\eta}\right)\right) \geq \delta \ .$$

Lemma 7. *Assume that ν contains at least 2 change points, $N = m \geq 2$. Then, there exists an environment ν' such that $H^{(m,\nu)}_{\mathrm{detect}} \leq H^{(m,\nu')}_{\mathrm{detect}} \leq 4H^{(m,\nu)}_{\mathrm{detect}}$, $\frac{1}{2}H^{(m,\nu)}_{\mathrm{localize}} \leq H^{(m,\nu')}_{\mathrm{localize}} \leq H^{(m,\nu)}_{\mathrm{localize}}$, and*

$$\mathbb{P}_{\pi,\nu'}\left(\mathcal{T}_\pi \geq \frac{1}{2}H^{(m,\nu)}_{\mathrm{detect}} \log\left(\frac{1}{6\delta}\right)\right) \geq \delta \ .$$

From these two lemmas, we can now deduce Theorem 2.

Proof of Theorem 2. The proof is a direct consequence of Lemmas 6 and 7.

First, assume that $N = m = 1$. Recall that in this case there is a single change point with energy $\mathcal{E}_1 = \Delta_1^2$ and sparsity $s_1 = 1$, so $H_{\mathrm{detect}} = H_{\mathrm{localize}}$. We apply Lemma 6 to obtain an environment ν' such that $H^{(m,\nu)}_{\mathrm{localize}} = H^{(m,\nu')}_{\mathrm{localize}}$ (and similarly for H_{detect}) and such that

$$\mathbb{P}_{\pi,\nu'}\left(\mathcal{T}_\pi \geq \frac{1}{\Delta_1^2} \log\left(\frac{1}{8\delta}\right) + \frac{1}{\Delta_1^2} \log_+\left(\frac{1}{16\eta}\right)\right) \geq \delta \ ,$$

and the claimed bound follows from the fact that $H_{\mathrm{detect}} = H_{\mathrm{localize}}$ in this case.

Second, assume that $N = m \geq 2$. We apply Lemma 7, and Lemma 6 to obtain an environment ν' such that $H^{(m,\nu)}_{\mathrm{detect}} \leq H^{(m,\nu')}_{\mathrm{detect}} \leq 4H^{(m,\nu)}_{\mathrm{detect}}$, $\frac{1}{2}H^{(m,\nu)}_{\mathrm{localize}} \leq H^{(m,\nu')}_{\mathrm{localize}} \leq H^{(m,\nu)}_{\mathrm{localize}}$, and $\mathbb{P}_{\pi,\nu'}(\mathcal{T}_\pi \geq \chi) \geq \delta$, with

$$\begin{aligned}\chi =& \max\left(\frac{1}{2}H^{(m,\nu)}_{\mathrm{detect}} \log\left(\frac{1}{6\delta}\right), H^{(m,\nu)}_{\mathrm{localize}} \log\left(\frac{1}{8\delta}\right) + \sum_{i=1}^{m} \frac{1}{\Delta_i^2} \log_+\left(\frac{s_i}{16\eta}\right)\right) \\ \geq& \frac{1}{4}H^{(m,\nu)}_{\mathrm{detect}} \log\left(\frac{1}{8\delta}\right) + \frac{1}{2}H^{(m,\nu)}_{\mathrm{localize}} \log\left(\frac{1}{8\delta}\right) + \frac{1}{2}\sum_{i=1}^{m} \frac{1}{\Delta_i^2} \log_+\left(\frac{s_i}{16\eta}\right) \ ,\end{aligned}$$

which concludes the proof. □

Now, we can deduce Corollary 1 from Theorem 2, and Theorem 5.2 from Lazzaro and Pike-Burke [2025a].

Proof of Corollary 1. First, observe that for any environment ν with $m \geq 1$ change points, the condition $\vartheta_i > 2\eta$ for $i = 2, \dots, m-1$ allows us to apply Theorem 5.2 from Lazzaro and Pike-Burke [2025a], which states that for any (δ, η, m)-correct algorithm π,

$$\mathbb{E}_{\pi,\nu}[\mathcal{T}_\pi] \geq 4H^{(m,\nu)}_{\mathrm{localize}} \log\left(\frac{1}{4\delta}\right) \ . \tag{31}$$

Second, we obtain the lower bound in $\mathbb{E}[\mathcal{T}_\pi]$ independent of δ by applying Theorem 2 with $\delta = 1/16$, observing that ν is in particular $(1/16, \eta, m)$-correct, as $\delta \leq 1/16$. From Theorem 2, there exists an environment ν' such that $H^{(m,\nu)}_{\mathrm{detect}} \leq H^{(m,\nu')}_{\mathrm{detect}} \leq 4H^{(m,\nu)}_{\mathrm{detect}}$, $\frac{1}{2}H^{(m,\nu)}_{\mathrm{localize}} \leq H^{(m,\nu')}_{\mathrm{localize}} \leq H^{(m,\nu)}_{\mathrm{localize}}$, and such that

$$\mathbb{P}_{\pi,\nu'}\left(\mathcal{T}_\pi \geq \tfrac{1}{4}H^{(m,\nu)}_{\mathrm{detect}}\log(2) + \tfrac{1}{2}H^{(m,\nu)}_{\mathrm{localize}}\log(2) + \tfrac{1}{2}\sum_{i=1}^{m}\frac{1}{\Delta_i^2}\log_+\left(\frac{s_i}{16\eta}\right)\right) \geq 1/16 \ .$$

Now, from Markov's inequality, we can lower bound $\mathbb{E}_{\pi,\nu'}[\mathcal{T}_\pi]$ as

$$\mathbb{E}_{\pi,\nu'}[\mathcal{T}_\pi] \geq \frac{1}{16}\left(\frac{1}{4}H^{(m,\nu)}_{\mathrm{detect}}\log(2) + \frac{1}{2}H^{(m,\nu)}_{\mathrm{localize}}\log(2) + \frac{1}{2}\sum_{i=1}^{m}\frac{1}{\Delta_i^2}\log_+\left(\frac{s_i}{16\eta}\right)\right) \tag{32}$$

Now, the final lower bound from Corollary 1 follows from Equations (31),(32), and the standard inequality $\max(a,b) \geq (a+b)/2$ for $a, b \geq 0$.

□

E.2 Proof of Lemma 6

Proof of Lemma 6. **Proof roadmap.** The proof has four steps.

First, we construct a family $(\nu_J)_J$, indexed by $J = (j_1, \ldots, j_m) \in \prod_{i=1}^m \{0, \ldots, \alpha_i - 1\}$, by shifting each change point x_i^* over $\alpha_i \sim s_i/8\eta$ candidate positions. We pay attention to preserve roughly the length of the change points, so that the complexities $H^{(m,\nu)}_{\mathrm{detect}}$ is preserved (up to a numerical constant) under any of the considered shifts. Second, for each $i \in \{1, \ldots, m\}$, we define an alternative environment $\nu_{J^{(-i)}}$ by removing from ν_J the i-th jump. Third, we compare the behavior of the algorithm under ν_J and $\nu_{J^{(-i)}}$. For that, we consider the event $\xi_{i,j_i} = \{\mathcal{T}_\pi \leq \chi,\ |\hat{x}_i - x_i^J| \leq \eta\}$, where x_i^J is the position of the i-th change point in environment ν_J, and χ is the largest $(1-\delta)$-quantile of the budget under the family of environments $(\nu_J)_J$. We show that ξ_{i,j_i} is likely under ν_J and unlikely on average under $\nu_{J^{(-i)}}$. These bounds allow us to lower bound the KL between the respective probability distributions, thanks to data-processing inequalities. Finally, we compute the KL divergence between the probability under ν_J and $\nu_{J^{(-i)}}$, which leads to a lower bound on the quantile budget χ, which translates Lemma 6.

Reference class of environments. Consider the parameters of the environment ν given by the jumps $\Delta_1, \ldots, \Delta_m$, the initial mean μ_0, and the change point positions $x_1^*, \ldots, x_m^*$, and the function f defined by Equation (3). Recall that $x_0^* = 0$ and $x_{m+1}^* = 1$. Consider the length of the change points $\vartheta_0 = 1$, for $i = 1, \ldots, m-1$ $\vartheta_i = x_{i+1}^* - x_i^*$, and $\vartheta_m = 1$.

Fix $\eta' > \eta$, which will be used to shift the change points. For $i = 1, \ldots, m$, define

$$\alpha_i := 1 \vee \left\lfloor \frac{s_i}{8\eta'} \right\rfloor \ ,$$

as the number of possible positions for the change point i in the environments that we will construct. Observe that $\alpha_i \geq 1$ for all $i = 1, \ldots, m$.

Let $J = (j_1, \ldots, j_m)$ be a vector of size m, so that for each $i = 1, \ldots, m$, $0 \leq j_i \leq \alpha_i - 1$. For each $i = 1, \ldots, m$, define

$$x_i^J = \frac{1}{2}x_i^* + j_i \cdot 2\eta' \tag{33}$$

By convention, $x_0^J := 0$ and $x_{m+1}^J := 1$.

For each $J = (j_1, \ldots, j_m)$, define ν_J as the environment with the same jumps as ν, and where the position of the change points are given by the vector $(x_i^J)_{i=1,\ldots,m}$. Note that ν_J has also m change points and that the jumps are unchanged. Moreover, for each $i = 1, \ldots, m-1$, the change points x_i^J and x_{i+1}^J are spaced by at least $\vartheta_i/4$ and at most $(3/4)\vartheta_i$, so that the complexities $H^{(m,\nu)}_{\mathrm{detect}}$ are preserved (up to a numerical constant) under any of the considered shifts. We point to Lemma 8 for a formal statement of this fact.

Denote as $\mathbb{P}_{\pi,J}$ the probability distribution induced by the interaction between an algorithm π, and the environment is ν_J, and by $\mathbb{E}_{\pi,J}$ the associated expectation.

Alternative environments. Let $i \in \{1, \dots, m\}$ and fix for now $j_i \in \{0, \dots, \alpha_i - 1\}$. Fix any $J \in \prod_{k=1}^{m} \{0, \dots, \alpha_k - 1\}$, such that j_i is the i-th component of J. Denote as $J^{(-i)}$ the vector obtained from J by removing the i-th component j_i. Define $\nu_{J^{(-i)}}$ as the environment obtained from ν_J by setting the jump Δ_i to zero, so that $\nu_{J^{(-i)}}$ has $m-1$ change points. Note that $\nu_{J^{(-i)}}$ and ν_J only differ for arms in the interval $[x_i^J, x_{i+1}^J)$, and are identical elsewhere. Observe moreover that $\nu_{J^{(-i)}}$ does not depend on the index j_i, as the jump at change point i is set to zero.

Denote as $\mathbb{P}_{\pi, J^{(-i)}}$ the probability distribution of the observations when the algorithm is π, and the environment is $\nu_{J^{(-i)}}$ and by $\mathbb{E}_{\pi, J^{(-i)}}$ the associated expectation.

Bound on total variation. Let π be an algorithm for this problem, which is (δ, η)-correct, and let $\hat{x}_1, \dots, \hat{x}_m$ be the corresponding estimates of the change points. Denote as $\mathcal{T}_\pi$ the stopping time of π. Define χ as the supremum of the $(1-\delta)$-quantile of the budget over the family of probability distributions $\mathbb{P}_{\pi, J}$ for $J \in \prod_{i=1}^{m} \{0, \dots, \alpha_i - 1\}$. Namely,

$$\chi = \inf\{t > 0 : \forall J \in \prod_{i=1}^{m} \{0, \dots, \alpha_i - 1\}, \mathbb{P}_{\pi, J}(\mathcal{T}_\pi \geq t) \leq \delta\} \ . \tag{34}$$

Let $i \in \{1, \dots, m\}$. Let J be a vector of m indices, whose i-th component is j_i. Define the event

$$\xi_{i, j_i} = \{\mathcal{T}_\pi \leq \chi, \ |\hat{x}_i - x_i^J| \leq \eta\} \ .$$

Recall that the environment ν_J contains m change points, and the i-th change point is x_i^J. Then, it follows from the (δ, η)-correctness of π that

$$\mathbb{P}_{\pi, J}(|\hat{x}_i - x_i^J| > \eta) \leq \delta \ .$$

Moreover, by definition of χ (see (34)), it holds that

$$\mathbb{P}_{\pi, J}(\mathcal{T}_\pi \geq \chi) \leq \delta \ .$$

Therefore, by union bound, we have that, for all J such that j_i is the i-th component of J,

$$\mathbb{P}_{\pi, J}(\xi_{i, j_i}) \geq 1 - 2\delta \ . \tag{35}$$

Consider now the environment $\nu_{J^{(-i)}}$, which differs from ν_J by the fact that the jump at change point i is set to zero. Under this environment, the change point i is not identifiable, and the algorithm should not be able to identify it in finite time. In particular, we have the following bound, stated as Lemma 9, whose proof is deferred at the end of this section, and which hold as long as $\eta < 1/8$.

$$\text{For all } J^{(-i)}, \text{ it holds that} \quad \mathbb{P}_{\pi, J^{(-i)}}(\mathcal{T}_\pi \leq \chi) \leq 2\delta \ . \tag{36}$$

Let $j_i \in \{0, \dots \alpha_i - 1\}$ and recall that the position of the change point i in the environment ν_J is $x_i^J = x_i^* + j_i \cdot 2\eta'$. In particular, for any $\tilde{j}_i \in \{0, \dots, \alpha_i - 1\}$, with $\tilde{j}_i \neq j_i$, if $\tilde{J}$ is such that its i-th component is $\tilde{j}_i$, then, one has $|x_i^J - x_i^{\tilde{J}}| \geq 2\eta' > 2\eta$. Therefore, the events ξ_{i, j_i} and $\xi_{i, \tilde{j}_i}$ are disjoint. Moreover, observe that the environment $\nu_{J^{(-i)}}$ does not depend on the index j_i, as the jump at change point i is set to zero.

Therefore, we can write, for a fixed i and for a fixed vector of indices $J^{(-i)}$,

$$\begin{aligned}
\frac{1}{\alpha_i} \sum_{j_i=0}^{\alpha_i - 1} \mathbb{P}_{\pi, J^{(-i)}}(\xi_{i, j_i}) &= \frac{1}{\alpha_i} \mathbb{E}_{\pi, J^{(-i)}} \left[\sum_{j_i=0}^{\alpha_i - 1} \mathbb{1}_{\xi_{i, j_i}} \right] = \frac{1}{\alpha_i} \mathbb{P}_{\pi, J^{(-i)}} \left[\bigsqcup_{j_i=0}^{\alpha_i - 1} \xi_{i, j_i} \right] \\
&\leq \frac{1}{\alpha_i} \mathbb{P}_{\pi, J^{(-i)}}(\mathcal{T}_\pi \leq \chi) \\
&\leq \frac{2\delta}{\alpha_i} \ ,
\end{aligned}$$

where we use that, $\forall j_i$, $\xi_{i, j_i} \subset \{\mathcal{T}_\pi \leq \chi\}$, and the last inequality follows from Equation (36).

On the other hand, it follows from Equation (35), averaging over j_i, that,

$$\frac{1}{\alpha_i}\sum_{j_i=0}^{\alpha_i-1}\mathbb{P}_{\pi,J}(\xi_{i,j_i}) \geq 1-2\delta \ ,$$

where we point out that the distribution $\mathbb{P}_{\pi,J}$ on the left sum depends on the index j_i.

From these two bounds, using monotonicity of $\mathrm{kl}(\cdot,\cdot)$ in each argument (valid as soon as $2\delta/\alpha_i \leq 1-2\delta$, which holds for $\delta \leq 1/4$ and $\alpha_i \geq 1$), and joint convexity of the kl divergence (e.g., Polyanskiy and Wu, 2014, Section 4.1), we write

$$\begin{aligned}\mathrm{kl}\left(1-2\delta, \frac{2\delta}{\alpha_i}\right) &\leq \mathrm{kl}\left(\frac{1}{\alpha_i}\sum_{j_i=0}^{\alpha_i-1}\mathbb{P}_{\pi,J}(\xi_{i,j_i}), \frac{1}{\alpha_i}\sum_{j_i=0}^{\alpha_i-1}\mathbb{P}_{\pi,J^{(-i)}}(\xi_{i,j_i})\right)\\ &\leq \frac{1}{\alpha_i}\sum_{j_i=0}^{\alpha_i-1}\mathrm{kl}\left(\mathbb{P}_{\pi,J}(\xi_{i,j_i}), \mathbb{P}_{\pi,J^{(-i)}}(\xi_{i,j_i})\right) \ . \end{aligned} \tag{37}$$

Observe that the bound (37) holds for any components $J^{(-i)} = (j_k)_{k\neq i}$. In particular, we can average over $J^{(-i)}$ to write

$$\begin{aligned}\mathrm{kl}\left(1-2\delta, \frac{2\delta}{\alpha_i}\right) &\leq \frac{1}{\alpha_i}\sum_{j_i=0}^{\alpha_i-1}\frac{1}{\prod_{k\neq i}\alpha_k}\sum_{J^{(-i)}\in\prod_{k\neq i}\{0,\dots,\alpha_k-1\}}\mathrm{kl}\left(\mathbb{P}_{\pi,J}(\xi_{i,j_i}), \mathbb{P}_{\pi,J^{(-i)}}(\xi_{i,j_i})\right)\\ &= \frac{1}{\prod_{k=1}^m\alpha_k}\sum_{J\in\prod_{k=1}^m\{0,\dots,\alpha_k-1\}}\mathrm{kl}\left(\mathbb{P}_{\pi,J}(\xi_{i,j_i}), \mathbb{P}_{\pi,J^{(-i)}}(\xi_{i,j_i})\right)\\ &= \mathbb{E}_{J\sim\mathcal{U}}\left[\mathrm{kl}\left(\mathbb{P}_{\pi,J}(\xi_{i,j_i}), \mathbb{P}_{\pi,J^{(-i)}}(\xi_{i,j_i})\right)\right] \ , \end{aligned}$$

where the last equality is the notation for the expectation over J when J is uniformly distributed in $\prod_{k=1}^m\{0,\dots,\alpha_k-1\}$.

Now, ξ_{i,j_i} is measurable with respect to the observations of π up to time χ, so that the probability of ξ_{i,j_i} under $\mathbb{P}_{\pi,J}$ and $\mathbb{P}_{\pi,J^{(-i)}}$ only depends on the distribution of the observations up to time χ. Consider then the procedure $\tilde{\pi}$ defined as follows: for $t=1,\dots,\chi$, $\tilde{\pi}$ behaves like π. If at time χ, the algorithm π did not stop, then $\tilde{\pi}$ stops anyway and returns an error. For any environment, the distribution of the observations of $\tilde{\pi}$ up to time χ is the same as the distribution of the observations of π up to time χ. Moreover, the algorithm $\tilde{\pi}$ stops at time χ at the latest, so that the budget of $\tilde{\pi}$ is at most χ. Therefore, we can rewrite the above inequality, and use the data-processing inequality for the KL divergence, to write

$$\mathrm{kl}\left(1-2\delta, \frac{2\delta}{\alpha_i}\right) \leq \mathbb{E}_{J\sim\mathcal{U}}\left[\mathrm{KL}\left(\mathbb{P}_{\tilde{\pi},J}, \mathbb{P}_{\tilde{\pi},J^{(-i)}}\right)\right] \ . \tag{38}$$

Bound on the KL **divergence.** Fix $i\in\{1,\dots,m\}$ and $J\in\prod_{k=1}^m\{0,\dots,\alpha_k-1\}$. We want to bound the KL divergence $\mathrm{KL}\left(\mathbb{P}_{\tilde{\pi},J}, \mathbb{P}_{\tilde{\pi},J^{(-i)}}\right)$.

Now, observe that the environments ν_J and $\nu_{J^{(-i)}}$ only differ by a constant magnitude Δ_i on the interval $[x_i^J, x_{i+1}^J)$, and are identical elsewhere. Denote as f_J for the reward function characterizing ν_J, and $f_{J^{(-i)}}$ for the reward function characterizing $\nu_{J^{(-i)}}$. One has

$$f_J - f_{J^{(-i)}} = \Delta_i \mathbb{1}\{x\in[x_i^J, x_{i+1}^J)\} \ .$$

Recall that the noise is assumed to be Gaussian with variance 1.

Therefore, we can use the decomposition of the KL divergence for bandit problems, under a continuous action space, to write

$$\begin{aligned}\mathrm{KL}\left(\mathbb{P}_{\tilde{\pi},J}, \mathbb{P}_{\tilde{\pi},J^{(-i)}}\right) &= \mathbb{E}_{\tilde{\pi},J}\left[\sum_{t=1}^{\mathcal{T}_{\tilde{\pi}}} \mathrm{KL}\left(\mathcal{N}(f_J(x_t),1), \mathcal{N}(f_{J^{(-i)}}(x_t),1)\right)\right] \\ &= \mathbb{E}_{\tilde{\pi},J}\left[\sum_{t=1}^{\mathcal{T}_{\tilde{\pi}}} \frac{\left(f_J(x_t) - f_{J^{(-i)}}(x_t)\right)^2}{2}\right] \\ &= \frac{\Delta_i^2}{2}\mathbb{E}_{\tilde{\pi},J}\left[\sum_{t=1}^{\mathcal{T}_{\tilde{\pi}}} \mathbb{1}\{x_t \in [x_i^J, x_{i+1}^J)\}\right] \ .\end{aligned}$$

Now, averaging over J uniformly distributed in $\prod_{k=1}^m \{0, \dots, \alpha_k - 1\}$, from Equation (38), we obtain

$$\frac{2}{\Delta_i^2}\,\mathrm{kl}\left(1-2\delta, \frac{2\delta}{\alpha_i}\right) \le \mathbb{E}_{J\sim\mathcal{U}}\left[\mathbb{E}_{\tilde{\pi},J}\left[\sum_{t=1}^{\mathcal{T}_{\tilde{\pi}}} \mathbb{1}\{x_t \in [x_i^J, x_{i+1}^J)\}\right]\right] \ .$$

Observe that the averaging measure $\mathbb{E}_{J\sim\mathcal{U}}\mathbb{E}_{\tilde{\pi},J}$ is the same for each index i. Then, we sum over $i = 1, \dots, m$

$$\begin{aligned}\sum_{i=1}^m \frac{2}{\Delta_i^2}\,\mathrm{kl}\left(1-2\delta, \frac{2\delta}{\alpha_i}\right) &\le \sum_{i=1}^m \mathbb{E}_{J\sim\mathcal{U}}\left[\mathbb{E}_{\tilde{\pi},J}\left[\sum_{t=1}^{\mathcal{T}_{\tilde{\pi}}} \mathbb{1}\{x_t \in [x_i^J, x_{i+1}^J)\}\right]\right] \\ &= \mathbb{E}_{J\sim\mathcal{U}}\left[\mathbb{E}_{\tilde{\pi},J}\left[\sum_{t=1}^{\mathcal{T}_{\tilde{\pi}}}\sum_{i=1}^m \mathbb{1}\{x_t \in [x_i^J, x_{i+1}^J)\}\right]\right] \\ &\le \mathbb{E}_{J\sim\mathcal{U}}\left[\mathbb{E}_{\tilde{\pi},J}\left[\mathcal{T}_{\tilde{\pi}}\right]\right] \\ &\le \chi \ ,\end{aligned}$$

where the second inequality follows from the fact that the intervals $[x_i^J, x_{i+1}^J)$ are disjoint, and the last inequality follows from the fact that, under $\tilde{\pi}$, the budget $\mathcal{T}_{\tilde{\pi}}$ is at most χ.

By definition of χ as an infimum, this means that there exists $J \in \prod_{i=1}^m \{0, \dots, \alpha_i - 1\}$ such that

$$\mathbb{P}_{\pi,J}\left(\mathcal{T}_\pi \ge 2\sum_{i=1}^m \Delta_i^{-2}\,\mathrm{kl}\left(1-2\delta, \frac{2\delta}{\alpha_i}\right)\right) \ge \delta \ ,$$

in particular, there exists an environment $\nu' = \nu_J$ such that the above holds. Moreover, by construction, the jumps of ν' are the same as the jumps of ν, and for each $i \in \{1, \dots, m-1\}$, $(1/4)\vartheta_i \le \vartheta_i' \le (3/4)\vartheta_i$, as verified in Lemma 8.

Finally, use the bound $\mathrm{kl}\left(1-2\delta, \frac{2\delta}{\alpha_i}\right) \ge \frac{1}{2}\log\left(\frac{\alpha_i}{8\delta}\right)$ from technical Lemma 10, and with the bound $\log(\alpha_i) \ge \log_+\left(\frac{s_i}{16\eta'}\right)$, which follows from the definition of α_i. As $\eta' > \eta$ is arbitrary, we can conclude by taking the limit $\eta' \to \eta$. □

Lemma 8. *For all $J \in \prod_{i=1}^m \{0, \dots, \alpha_i - 1\}$, it holds that ν_J is a valid environment with m change points, with the same jumps at ν, and with spacing $(1/4)\vartheta_i \le \vartheta_i^J \le (3/4)\vartheta_i$ for each $i = 1, \dots, m-1$. In particular, $H^{\nu_J}_{\mathrm{localize}} = H^{\nu}_{\mathrm{localize}}$, and $\frac{4}{3}H^{\nu}_{\mathrm{detect}} \le H^{\nu_J}_{\mathrm{detect}} \le 4H^{\nu}_{\mathrm{detect}}$.*

Proof. First, we verify that the environment ν_J is valid and contains m change points. For validity, we only need the positions of the change points $(x_i^J)_{i=1}^m$ to be distinct and to belong to $(0,1)$.

Consider $x_m^J = \frac{1}{2}x_m^* + j_m \cdot 2\eta'$. One has $j_m \cdot 2\eta' \le \alpha_m \cdot 2\eta' \le \frac{s_m}{8\eta'} \cdot 2\eta' \le 1/4$, and $x_m^* \le 1$, so that $x_m^J < 1$.

Moreover, for each $i = 1, \dots, m-1$, the change points x_i^J and x_{i+1}^J are spaced by at least $\vartheta_i/4$ and at most $(3/4)\vartheta_i$. Indeed, one has $x_{i+1}^J - x_i^J = \frac{1}{2}(x_{i+1}^* - x_i^*) + (j_{i+1} - j_i)\cdot 2\eta' = \frac{1}{2}\vartheta_i + (j_{i+1} - j_i)\cdot 2\eta'$. We will show that $|j_{i+1} - j_i| \cdot 2\eta' \le \frac{\vartheta_i}{4}$, which will conclude the proof.

By assumption on J, we have that $j_{i+1} - j_i \in \{-(\alpha_i - 1), \dots, \alpha_{i+1} - 1\}$, so that $|j_{i+1} - j_i| \cdot 2\eta' \leq ((\alpha_i - 1) \vee (\alpha_{i+1} - 1)) \cdot 2\eta'$. Now, by definition of α_i and α_{i+1}, we have $\alpha_i \vee \alpha_{i+1} - 1 = 1 \vee \left\lfloor \frac{\vartheta_{i-1} \wedge \vartheta_i}{8\eta'} \right\rfloor \vee \left\lfloor \frac{\vartheta_i \wedge \vartheta_{i+1}}{8\eta'} \right\rfloor - 1 \leq 0 \vee \left(\frac{\vartheta_i}{8\eta'} - 1 \right)$. Therefore, $|j_{i+1} - j_i| \cdot 2\eta' \leq 0 \vee \left(\frac{\vartheta_i}{4} - 2\eta' \right) \leq \frac{\vartheta_i}{4}$, which conclude the proof.

□

Lemma 9.

For all $J^{(-i)}$, it holds that $\mathbb{P}_{\pi, J^{(-i)}}(\mathcal{T}_\pi \leq \chi) \leq 2\delta$.

Proof of Lemma 9. We can prove it by considering two environments, with one vanishing change point of jump magnitude ϵ. Denote as $\nu_\epsilon^{(1)}$ the environment obtained from $\nu_{J^{(-i)}}$ by adding a signal of magnitude ϵ on the interval $[x_m^* + l', 1)$, and as $\nu_\epsilon^{(2)}$ the environment obtained from $\nu_{J^{(-i)}}$ by adding a signal of magnitude ϵ on the interval $[1 - l', 1)$. Here, we take $0 < l' < 1/8 - \eta$, which is possible by assumption on $\eta < 1/8$. These choice imply that the m-th change point under $\nu_\epsilon^{(1)}$, and the m-th change point under $\nu_\epsilon^{(2)}$ are spaced by at least 2η. Indeed $x_m^* + l' + \eta < 1 - l' - \eta$, which hold for $x_m^* \leq 3/4 \leq 1 - l' - 2\eta$. Note that $\nu_\epsilon^{(1)}$ and $\nu_\epsilon^{(2)}$ have m valid change points.

Consider the decomposition of the event $\{\mathcal{T}_\pi \leq \chi\}$ as

$$\{\mathcal{T}_\pi \leq \chi\} = \{\mathcal{T}_\pi \leq \chi,\ \hat{x}_m \leq \frac{1}{2}(x_m^* + 1)\} \cup \{\mathcal{T}_\pi \leq \chi,\ \hat{x}_m > \frac{1}{2}(x_m^* + 1)\} \ . \tag{39}$$

Under $\nu_\epsilon^{(1)}$, the change point m is at position $x_m^* + l' < \frac{1}{2}(x_m^* + 1) - \eta$. Then, by the (δ, η)-correctness of π, it holds that

$$\mathbb{P}_{\pi, \nu_\epsilon^{(1)}}\left(\hat{x}_m > \frac{1}{2}(x_m^* + 1) \right) \leq \mathbb{P}_{\pi, \nu_\epsilon^{(1)}}(|\hat{x}_m - (x_m^* + l')| > \eta) \leq \delta \ . \tag{40}$$

Similarly, under $\nu_\epsilon^{(2)}$, the change point m is at position $1 - l' > \frac{1}{2}(x_m^* + 1) + \eta$, so that

$$\mathbb{P}_{\pi, \nu_\epsilon^{(2)}}\left(\hat{x}_m \leq \frac{1}{2}(x_m^* + 1) \right) \leq \mathbb{P}_{\pi, \nu_\epsilon^{(2)}}(|\hat{x}_m - (1 - l')| > \eta) \leq \delta \ . \tag{41}$$

Moreover, when ϵ goes to zero, the total variation distance between $\mathbb{P}_{\pi, \nu_\epsilon^{(1)}}$ and $\mathbb{P}_{\pi, J^{(-i)}}$, goes to zero when we restrict to events measurable with respect to the observations up to the finite time χ. Consider any event A measurable with respect to the observations up to time χ. Then, we can write, by decomposition of the KL divergence,

$$\begin{aligned}
\mathrm{KL}(\mathbb{P}_{\pi, \nu_\epsilon^{(1)}}(A), \mathbb{P}_{\pi, J^{(-i)}}(A)) &\leq \mathbb{E}_{\pi, \nu_\epsilon^{(1)}}\left[\sum_{t=1}^{\chi} \mathrm{KL}(\mathcal{N}(f_\epsilon^{(1)}(x_t), 1), \mathcal{N}(f_{J^{(-i)}}(x_t), 1)) \right] \\
&= \frac{\epsilon^2}{2} \mathbb{E}_{\pi, \nu_\epsilon^{(1)}}\left[\sum_{t=1}^{\chi} \mathbb{1}\{x_t \in [x_m^* + l', 1 - l'\} \right] \\
&\leq \frac{\epsilon^2}{2} \chi \ ,
\end{aligned}$$

Now, from the Pinsker's inequality, we have that

$$|\mathbb{P}_{\pi, \nu_\epsilon^{(1)}}(A) - \mathbb{P}_{\pi, J^{(-i)}}(A)| \leq \sqrt{\frac{1}{2} \mathrm{KL}(\mathbb{P}_{\pi, \nu_\epsilon^{(1)}}(A), \mathbb{P}_{\pi, J^{(-i)}}(A))} \leq \epsilon \sqrt{\chi}/2 \ ,$$

which goes to zero as ϵ goes to zero. We proceed similarly for $\nu_\epsilon^{(2)}$.

Now, using the decomposition of the event $\{\mathcal{T}_\pi \leq \chi\}$ given by Equation (39), we can write

$$\begin{aligned}
\mathbb{P}_{\pi, J^{(-i)}}(\mathcal{T}_\pi \leq \chi) &= \mathbb{P}_{\pi, J^{(-i)}}(\mathcal{T}_\pi \leq \chi,\ \hat{x}_m \leq \tfrac{1}{2}(x_m^* + 1)) + \mathbb{P}_{\pi, J^{(-i)}}(\mathcal{T}_\pi \leq \chi,\ \hat{x}_m > \tfrac{1}{2}(x_m^* + 1)) \\
&= \lim_{\epsilon \to 0} \mathbb{P}_{\pi, \nu_\epsilon^{(2)}}(\mathcal{T}_\pi \leq \chi,\ \hat{x}_m \leq \tfrac{1}{2}(x_m^* + 1)) \\
&\quad + \lim_{\epsilon \to 0} \mathbb{P}_{\pi, \nu_\epsilon^{(1)}}(\mathcal{T}_\pi \leq \chi,\ \hat{x}_m > \tfrac{1}{2}(x_m^* + 1)) \\
&\leq \lim_{\epsilon \to 0} \mathbb{P}_{\pi, \nu_\epsilon^{(2)}}(\hat{x}_m \leq \tfrac{1}{2}(x_m^* + 1)) + \lim_{\epsilon \to 0} \mathbb{P}_{\pi, \nu_\epsilon^{(1)}}(\hat{x}_m > \tfrac{1}{2}(x_m^* + 1)) \\
&\leq 2\delta \ ,
\end{aligned}$$

where the last inequality follows from the bounds (40),(41) we obtained under $\nu_\epsilon^{(1)}$ and $\nu_\epsilon^{(2)}$. This proves Equation (36). □

Lemma 10. *For $\delta \in (0, 1/4)$ and $\alpha \geq 1$, one has*

$$\mathrm{kl}\left(1-2\delta, \frac{2\delta}{\alpha}\right) \geq \frac{1}{2}\log\left(\frac{\alpha}{8\delta}\right) \quad .$$

Proof of Lemma 10. We can write

$$\begin{aligned}
\mathrm{kl}\left(1-2\delta, \frac{2\delta}{\alpha}\right) &\geq \mathrm{kl}\left(1/2, \frac{2\delta}{\alpha}\right) \\
&= \frac{1}{2}\log\left(\frac{1/2}{2\delta/\alpha}\right) + \frac{1}{2}\log\left(\frac{1/2}{1-2\delta/\alpha}\right) \\
&= \frac{1}{2}\log\left(\frac{\alpha}{8\delta}\right) + \frac{1}{2}\log\left(\frac{1}{(1-2\delta/\alpha)}\right) \\
&\geq \frac{1}{2}\log\left(\frac{\alpha}{8\delta}\right) \quad ,
\end{aligned}$$

where the first inequality follows from the fact that $1 - 2\delta \geq 1/2 \geq 2\delta/\alpha$, and the last inequality follows from the fact that $1 - 2\delta/\alpha \leq 1$. □

E.3 Proof of Lemma 7

Proof roadmap. The proof proceeds in four steps. The overall novelty lies in a two-level reduction: from bandit detection of multiple change points to an instance with two change points, then from the δ-PAC bandit setting with quantile guarantees to a batch detection problem amenable to classical information-theoretic tools.

First, we construct a family of environments $(\tilde{\nu}^{(s)})_{s\in[0,1/4]}$ from ν, and show that H_{detect} and H_{localize} are preserved up to absolute constants (Lemma 11). Second, we introduce an alternative environment $\bar{\nu}$ (with two removed jumps). Third, we consider the event $\xi = \{\mathcal{T}_\pi \leq \chi\}$, where χ is the maximum $(1-\delta)$-quantile of the budget under all the environments $\tilde{\nu}^{(s)}$. This event has a probability at least δ under $\nu' = \tilde{\nu}^{(s)}$ for all $s \in [0, 1/4]$. We show in Lemma 12 that it has a probability smaller than 2δ under $\bar{\nu}$, because $\bar{\nu}$ contains less than $m = N$ change points, so that the algorithm should not stop. Fourth, we convert this total-variation lower bound into a KL lower bound using the Bretagnolle–Huber inequality. Finally, we upper-bound the averaged KL by the expected number of pulls in the interval where $\bar{\nu}$ and $\tilde{\nu}^{(s)}$ differ, which yields a lower bound on χ; the conclusion follows by selecting $\nu' = \tilde{\nu}^{(s)}$ for a suitable s.

Proof of Lemma 7 . **Reference class of environments.** Consider an environment ν with $m \geq 2$ change points, with parameters $\mu_0, \Delta_1, \ldots, \Delta_m, \vartheta_0, \ldots, \vartheta_m$.

Observe that,

$$H^\nu_{\mathrm{detect}} := H^{(m,\nu)}_{\mathrm{detect}} = \max_{i=1,\ldots,m} \frac{1}{\mathcal{E}_i^2} = \max_{i=1,\ldots,m-1} \frac{1}{\vartheta_i}\frac{1}{\Delta_i^2 \wedge \Delta_{i+1}^2} \quad .$$

The quantity $\frac{1}{\vartheta_i}\frac{1}{\Delta_i^2\wedge\Delta_{i+1}^2}$ measures the difficulty of detecting a point in the interval between x_i^* and x_{i+1}^*. Consider $i^* = \arg\max_{i=1,\ldots,m-1} \frac{1}{\vartheta_i}\frac{1}{\Delta_i^2\wedge\Delta_{i+1}^2}$, and assume without loss of generality[10] that $|\Delta_{i^*}| \leq |\Delta_{i^*+1}|$, so that $H^\nu_{\mathrm{detect}} = \frac{1}{\vartheta_{i^*}}\frac{1}{\Delta_{i^*}^2}$.

From ν, we construct a family of environments $\tilde{\nu}^{(s)}$ parametrized by some $s \in [0, 1/4]$. Define $\tilde{\Delta} = \sqrt{2}(|\Delta_{i^*}| \wedge |\Delta_{i^*+1}|) = \sqrt{2}|\Delta_{i^*}|$ and $\tilde{\vartheta} = \vartheta_{i^*}/2$.

Let $s \in [0, 1/4]$, define the environment $\tilde{\nu}^{(s)}$ with parameters $\tilde{\Delta}_1, \ldots, \tilde{\Delta}_m$, the initial mean μ_0 and the length of the change points denoted as $\tilde{\vartheta}_0^{(s)}, \ldots, \tilde{\vartheta}_m^{(s)}$, where

[10] The other case follows from a symmetric construction

- the initial mean is μ_0 and initial position is $\tilde{x}_1^* = x_1^*/2$;
- for $i \in \{1, \dots, m\} \setminus \{i^*, i^*+1\}$, $\tilde{\Delta}_i = \Delta_i$;
- $\tilde{\Delta}_{i^*} = -\tilde{\Delta}_{i^*+1} = \tilde{\Delta}$;
- for $i \in \{1, \dots, m-1\} \setminus \{i^*-1, i^*, i^*+1\}$, $\tilde{\vartheta}_i^{(s)} = \vartheta_i/2$;
- $\tilde{\vartheta}_{i^*}^{(s)} = \vartheta_{i^*}/2$; $\tilde{\vartheta}_{i^*-1}^{(s)} = \vartheta_{i^*-1}/2 + s$ and $\tilde{\vartheta}_{i^*+1}^{(s)} = \vartheta_{i^*+1}/2 + \frac{1}{2} - s$

For notational purposes, denote as $\tilde{l} = \sum_{j=0}^{i^*-1} \tilde{\vartheta}_j^{(0)}$ for the position of the change point i^* under $\tilde{\nu}^{(0)}$. Observe that by definition, $\tilde{l} = \sum_{j=0}^{i^*-1} \vartheta_j/2 = x_{i^*}^*/2$. Recall that $\tilde{\vartheta} = \vartheta_{i^*}/2$. Define also $\tilde{r} = \tilde{l} + \tilde{\vartheta} + 1/4$ for the position of the change point $i^* + 1$ under $\tilde{\nu}^{(1/4)}$. Observe that $\tilde{r} = \sum_{j=0}^{i^*} \tilde{\vartheta}^{(1/4)} = \sum_{j=0}^{i^*} \vartheta_j/2 + 1/4 = x_{i^*+1}^*/2 + 1/4$. Moreover, one has $|\tilde{r} - \tilde{l}| = \tilde{\vartheta} + 1/4$. We denote as $\mathbb{P}_{\pi,s}$ the probability distribution induced by the interaction between algorithm π and the environment $\tilde{\nu}^{(s)}$. We will also denote by $\mathbb{E}_{\pi,s}$ the corresponding expectation.

Remark 2. *The family of environments $\left(\tilde{\nu}^{(s)}\right)_s$ is designed to preserve, up to a multiplicative constant 2 the complexities H_{detect} and H_{localize}, that is $H_{\mathrm{detect}}^{\tilde{\nu}^{(s)}}$ and $H_{\mathrm{localize}}^{\tilde{\nu}^{(s)}}$ are of the same order as $H_{\mathrm{detect}}^{\nu}$ and $H_{\mathrm{localize}}^{\nu}$, respectively. This is proved in Lemma 11 and is a direct consequence of the construction of $\tilde{\nu}^{(s)}$ and the definition of H_{detect} and H_{localize}.*

Alternative environment. Define $\bar{\nu}$ as the environment obtained from $\tilde{\nu}^{(0)}$ by setting the jumps $\tilde{\Delta}_{i^*}$ and $\tilde{\Delta}_{i^*+1}$ to zero, so that $\bar{\nu}$ has $m-2$ change points. Note that $\bar{\nu}$ does not depend on the parameter s. Interestingly, the environment $\tilde{\nu}^{(s)}$ and $\bar{\nu}$ only differ for arms in the interval $[\tilde{l}, \tilde{r}]$.

Denote as $\bar{\mathbb{P}}_\pi$ the probability distribution of the observations when the environment is $\bar{\nu}$ and by $\bar{\mathbb{E}}_\pi$ the associated expectation, with some algorithm π.

Remark 3. *This construction is a reduction scheme. We reduce the problem of detecting m change points to the signal detection problem of detecting the presence of a signal of magnitude $\tilde{\Delta}$, on an interval of length $\tilde{\vartheta}$, planted somewhere in the interval $[\tilde{l}, \tilde{r}]$.*

Bound in total variation. Let π be an algorithm, which is (δ, η)-correct for the change point detection problem with m change points, and let $\hat{x}_1, \dots, \hat{x}_m$ be the corresponding estimates of the change points. Denote as $\mathcal{T}_\pi$ the stopping time of π. Define χ as the supremum of the $(1-\delta)$-quantile of the budget over the family of probability distributions $\mathbb{P}_{\pi,s}$ for $s \in [0, 1/4]$. Namely,

$$\chi = \inf \left\{ t > 0 : \sup_{s \in [0,1/4]} \mathbb{P}_{\pi,s}(\mathcal{T}_\pi > t) \leq \delta \right\} \ .$$

Define the event

$$\xi = \{\mathcal{T}_\pi \leq \chi\} \ .$$

By definition of χ, it holds that $\mathbb{P}_{\pi,s}(\xi^c) \leq \delta$ for all $s \in [0, 1/4]$. Then, under $\bar{\nu}$, the algorithm is not able to identify m change points in finite time, as $\bar{\nu}$ only contains $m-2$ change points. In particular, one can prove the following bound, proved in Lemma 12 on the probability of the event ξ under $\bar{\nu}$:

$$\bar{\mathbb{P}}_\pi(\xi) \leq 2\delta \ . \tag{42}$$

This bound is a consequence of the fact that, under $\bar{\nu}$, the algorithm π should typically not stop, as it cannot identify m change points. In particular, the event ξ is an event on which π stops, and therefore should have a small probability under $\bar{\nu}$. The proof follows from proving that $\bar{\nu}$ is as close as possible to two different environments with m change points, under which the position of the change points are different.

Now, observe that the event ξ is measurable with respect to the first χ observations. Consider the modified algorithm $\tilde{\pi}$ defined as follows. For $t = 1, \dots, \chi$, $\tilde{\pi}$ uses the rules of π. If at time χ, the

algorithm π did not stop, $\tilde{\pi}$ stops anyway and returns an error. Observe that ξ is exactly the event on which $\tilde{\pi}$ does not return an error. In particular, the event ξ is measurable with respect to the observations of $\tilde{\pi}$, and has the same probability under π and $\tilde{\pi}$ (facing any environment). Therefore, $\bar{\mathbb{P}}_{\tilde{\pi}}(\xi) \leq 2\delta$ and $\mathbb{P}_{\tilde{\pi},s}(\xi) \geq 1-\delta$ for all $s \in [0, 1/4]$.

We can bound the total variation distance between $\bar{\mathbb{P}}_{\tilde{\pi}}$ and $\mathbb{P}_{\tilde{\pi},s}$ for $s \in [0, 1/4]$ as

$$\begin{aligned}\mathrm{TV}(\bar{\mathbb{P}}_{\tilde{\pi}}, \mathbb{P}_{\tilde{\pi},s}) &\geq \bar{\mathbb{P}}_{\tilde{\pi}}(\xi^c) - \mathbb{P}_{\tilde{\pi},s}(\xi^c) \\ &\geq 1 - 2\delta - \delta = 1 - 3\delta \ .\end{aligned}$$

Data processing inequality. Classically, one can bound the total variation distance between $\bar{\mathbb{P}}_{\tilde{\pi}}$ and $\mathbb{P}_{\tilde{\pi},s}$ by the Kullback–Leibler divergence between these two distributions, using the Bretagnolle–Huber inequality as

$$\mathrm{TV}(\bar{\mathbb{P}}_{\tilde{\pi}}, \mathbb{P}_{\tilde{\pi},s}) \leq 1 - \frac{1}{2}\exp\left(-\mathrm{KL}(\bar{\mathbb{P}}_{\tilde{\pi}}, \mathbb{P}_{\tilde{\pi},s})\right) \ .$$

Rearranging this inequality, and using the previous bound on the total variation distance, we obtain, for any $s \in [0, 1/4]$,

$$\mathrm{KL}(\bar{\mathbb{P}}_{\tilde{\pi}}, \mathbb{P}_{\tilde{\pi},s}) \geq \log\left(\frac{1}{2(1 - \mathrm{TV}(\bar{\mathbb{P}}_{\tilde{\pi}}, \mathbb{P}_{\tilde{\pi},s}))}\right) \geq \log\left(\frac{1}{6\delta}\right) \ . \tag{43}$$

Lower bound on the budget. Finally, we can bound the Kullback–Leibler divergence between $\bar{\mathbb{P}}_{\tilde{\pi}}$ and $\mathbb{P}_{\tilde{\pi},s}$, using the decomposition of the KL divergence for bandit problems, under a continuous action space. Observe that the environments $\bar{\nu}$ and $\tilde{\nu}^{(s)}$ only differ by a constant magnitude $\tilde{\Delta}$ on the interval $[\tilde{l}+s, \tilde{l}+s+\tilde{\vartheta})$, and are identical elsewhere. For notation, denote as $\bar{f}$ and $\tilde{f}^{(s)}$ the mean reward functions of $\bar{\nu}$ and $\tilde{\nu}^{(s)}$, respectively. Recall that the reward distributions are Gaussian with variance 1. Denote also as a_t the arm pulled at time t by $\tilde{\pi}$. Then, the reward distribution at time t under $\bar{\nu}$ is $\mathcal{N}(\bar{f}(a_t), 1)$ and the reward distribution at time t under $\tilde{\nu}^{(s)}$ is $\mathcal{N}(\tilde{f}^{(s)}(a_t), 1)$.

Therefore, we can write

$$\begin{aligned}\mathrm{KL}(\bar{\mathbb{P}}_{\tilde{\pi}}, \mathbb{P}_{\tilde{\pi},s}) &= \bar{\mathbb{E}}_{\tilde{\pi}}\left[\sum_{t=1}^{\mathcal{T}_{\tilde{\pi}}} \mathrm{KL}\left(\mathcal{N}(\bar{f}(a_t), 1), \mathcal{N}(\tilde{f}^{(s)}(a_t), 1)\right)\right] \\ &= \bar{\mathbb{E}}_{\tilde{\pi}}\left[\sum_{t=1}^{\mathcal{T}_{\tilde{\pi}}} \frac{\left(\tilde{f}^{(s)}(a_t) - \bar{f}(a_t)\right)^2}{2}\right] \\ &= \frac{\tilde{\Delta}^2}{2}\bar{\mathbb{E}}_{\tilde{\pi}}\left[\sum_{t=1}^{\mathcal{T}_{\tilde{\pi}}} \mathbb{1}\{a_t \in [\tilde{l}+s, \tilde{l}+s+\tilde{\vartheta})\}\right] \ .\end{aligned}$$

Now, averaging over s uniformly distributed in $[0, 1/4]$, we obtain

$$\begin{aligned}\frac{1}{1/4}\int_0^{1/4} \mathrm{KL}(\bar{\mathbb{P}}_{\tilde{\pi}}, \mathbb{P}_{\tilde{\pi},s})ds &= \frac{1}{1/4}\frac{\tilde{\Delta}^2}{2}\bar{\mathbb{E}}_{\tilde{\pi}}\left[\sum_{t=1}^{\mathcal{T}_{\tilde{\pi}}} \int_0^{1/4} \mathbb{1}\{a_t \in [\tilde{l}+s, \tilde{l}+s+\tilde{\vartheta})\}ds\right] \\ &= 2\tilde{\Delta}^2\bar{\mathbb{E}}_{\tilde{\pi}}\left[\sum_{t=1}^{\mathcal{T}_{\tilde{\pi}}} \int_0^{1/4} \mathbb{1}\{s \in [a_t - \tilde{l} - \tilde{\vartheta}, a_t - \tilde{l})\}ds\right] \\ &\leq 2\tilde{\vartheta}\tilde{\Delta}^2\bar{\mathbb{E}}_{\tilde{\pi}}\left[\mathcal{T}_{\tilde{\pi}}\right] \\ &,\end{aligned}$$

where the last inequality follows from the fact that the length of the interval $[a_t - \tilde{l} - \tilde{\vartheta}, a_t - \tilde{l})$ is at most $\tilde{\vartheta}$. Now, using the fact that, under $\tilde{\pi}$, the budget $\mathcal{T}_{\tilde{\pi}}$ is at most χ, we can write

$$\log\left(\frac{1}{6\delta}\right) \leq \frac{1}{1/4}\int_0^{1/4} \mathrm{KL}(\bar{\mathbb{P}}_{\tilde{\pi}}, \mathbb{P}_{\tilde{\pi},s})ds \leq 2\tilde{\vartheta}\tilde{\Delta}^2\chi \ , \tag{44}$$

where we used the bound on the KL divergence from Equation (43) in the first inequality.

Rearranging yields

$$\chi \geq \frac{1}{2\tilde{\vartheta}\tilde{\Delta}^2} \log\left(\frac{1}{6\delta}\right) = \frac{1}{2\vartheta_{i^*}} \frac{1}{\Delta_{i^*}^2 \wedge \Delta_{i^*+1}^2} \log\left(\frac{1}{6\delta}\right) = \frac{1}{2} H_{\mathrm{detect}}^{\nu} \log\left(\frac{1}{6\delta}\right) \ . \tag{45}$$

By definition of χ, this means that there exists $s \in [0, 1/4]$ such that

$$\mathbb{P}_{\pi,s}\left(\mathcal{T}_\pi \geq \frac{1}{2} H_{\mathrm{detect}}^{\nu} \log\left(\frac{1}{6\delta}\right)\right) \geq \delta \ .$$

The proof of Lemma 7 is concluded by observing that, by Lemma 11, the environment $\tilde{\nu}^{(s)}$ has the desired properties. It remains to prove Lemma 12 and Lemma 11. □

Lemma 11. *For all $s \in [0, 1/4]$ it holds that $\tilde{\nu}^{(s)}$ is a valid environment with m change points spaced by at least 2η and that*

$$H_{\mathrm{detect}}^{\nu} \leq H_{\mathrm{detect}}^{\tilde{\nu}^{(s)}} \leq 2H_{\mathrm{detect}}^{\nu} \quad \textit{and} \quad \frac{1}{2} H_{\mathrm{localize}}^{\nu} \leq H_{\mathrm{localize}}^{\tilde{\nu}^{(s)}} \leq H_{\mathrm{localize}}^{\nu} \ .$$

Proof of the Lemma 11. It is clear that $\tilde{\nu}^{(s)}$ is a valid environment with m change points, and that the change points are spaced by at least 2η, because $\tilde{\vartheta}_i^{(s)} \geq \vartheta_i/2 \geq 2\eta$ for all $i = 1, \ldots, m-1$.

By definition of $\tilde{\nu}^{(s)}$, it holds that $\tilde{\Delta}_{i^*} = -\tilde{\Delta}_{i^*+1} = \sqrt{2}\Delta_{i^*}$ and $\tilde{\vartheta}_{i^*}^{(s)} = \vartheta_{i^*}/2$. Then, one has

$$H_{\mathrm{localize}}^{\tilde{\nu}^{(s)}} = \sum_{i=1}^{m} \frac{1}{\tilde{\Delta}_i^2} = \sum_{\substack{i=1 \\ i \neq i^*, i^*+1}}^{m} \frac{1}{\Delta_i^2} + \frac{2}{\tilde{\Delta}^2} = \sum_{\substack{i=1 \\ i \neq i^*, i^*+1}}^{m} \frac{1}{\Delta_i^2} + \frac{1}{\Delta_{i^*}^2} \ .$$

Moreover, it holds that $\frac{1}{2}\left(\frac{1}{\Delta_{i^*}^2} + \frac{1}{\Delta_{i^*+1}^2}\right) \leq \frac{1}{\Delta_{i^*}^2} \leq \frac{1}{\Delta_{i^*}^2} + \frac{1}{\Delta_{i^*+1}^2}$, because $|\Delta_{i^*}| \leq |\Delta_{i^*+1}|$. Then, one has $\frac{1}{2} H_{\mathrm{localize}}^{\nu} \leq H_{\mathrm{localize}}^{\tilde{\nu}^{(s)}} \leq H_{\mathrm{localize}}^{\nu}$.

Recall that we assumed that $\Delta_{i^*} \leq \Delta_{i^*+1}$, so that

$$H_{\mathrm{detect}}^{\nu} = \max_{i=1,\ldots,m-1} \frac{1}{\vartheta_i} \frac{1}{\Delta_i^2 \wedge \Delta_{i+1}^2} = \frac{1}{\vartheta_{i^*}} \frac{1}{\Delta_{i^*}^2}$$
$$H_{\mathrm{detect}}^{\tilde{\nu}^{(s)}} = \max_{i=1,\ldots,m-1} \frac{1}{\tilde{\vartheta}_i^{(s)}} \frac{1}{\tilde{\Delta}_i^2 \wedge \tilde{\Delta}_{i+1}^2} \ .$$

We consider the different cases for $i = 1, \ldots, m-1$ to control $\frac{1}{\tilde{\vartheta}_i^{(s)}} \frac{1}{\tilde{\Delta}_i^2 \wedge \tilde{\Delta}_{i+1}^2}$.

Assume that $i \neq i^* - 1, i^*, i^* + 1$, so that $\tilde{\vartheta}_i^{(s)} = \vartheta_i/2$ and $\tilde{\Delta}_i = \Delta_i$ and $\tilde{\Delta}_{i+1} = \Delta_{i+1}$. Then, one has

$$\frac{1}{\tilde{\vartheta}_i^{(s)}} \frac{1}{\tilde{\Delta}_i^2 \wedge \tilde{\Delta}_{i+1}^2} = \frac{2}{\vartheta_i} \frac{1}{\Delta_i^2 \wedge \Delta_{i+1}^2} \leq 2H_{\mathrm{detect}}^{\nu} \ .$$

If $i = i^* - 1$, one has $\tilde{\vartheta}_{i^*-1}^{(s)} = \vartheta_{i^*-1}/2 + s$, $\tilde{\Delta}_{i^*-1} = \Delta_{i^*-1}$ and $\tilde{\Delta}_{i^*} = \sqrt{2}\Delta_{i^*}$. Then, one has

$$\frac{1}{\tilde{\vartheta}_{i^*-1}^{(s)}} \frac{1}{\tilde{\Delta}_{i^*-1}^2 \wedge \tilde{\Delta}_{i^*}^2} = \frac{1}{(\vartheta_{i^*-1}/2 + s)} \frac{1}{\Delta_{i^*-1}^2 \wedge \left(\sqrt{2}\Delta_{i^*}\right)^2} \leq \frac{2}{\vartheta_{i^*-1}} \frac{1}{\Delta_{i^*-1}^2 \wedge \Delta_{i^*}^2} \leq 2H_{\mathrm{detect}}^{\nu} \ ,$$

as $s \geq 0$ and $\sqrt{2}|\Delta_{i^*}| \geq |\Delta_{i^*}|$.

If $i = i^*$, one has $\tilde{\vartheta}_{i^*}^{(s)} = \vartheta_{i^*}/2$, $\tilde{\Delta}_{i^*} = \sqrt{2}\Delta_{i^*}$ and $\tilde{\Delta}_{i^*+1} = -\sqrt{2}\Delta_{i^*}$. Then,

$$\frac{1}{\tilde{\vartheta}_{i^*}^{(s)}} \frac{1}{\tilde{\Delta}_{i^*}^2 \wedge \tilde{\Delta}_{i^*+1}^2} = \frac{1}{(\vartheta_{i^*}/2)} \frac{1}{\left(\sqrt{2}\Delta_{i^*}\right)^2} = \frac{1}{\vartheta_{i^*}} \frac{1}{\Delta_{i^*}^2} = H_{\mathrm{detect}}^{\nu} \ .$$

Finally, if $i = i^* + 1$, one has $\tilde{\vartheta}^{(s)}_{i^*+1} = \vartheta_{i^*+1}/2 + 1/2 - s \geq \frac{1}{4}$ because $s \leq 1/4$. Also, $\tilde{\Delta}_{i^*+1} = -\sqrt{2}\Delta_{i^*}$ and $\tilde{\Delta}_{i^*+2} = \Delta_{i^*+2}$. Then,

$$\begin{aligned}\frac{1}{\tilde{\vartheta}^{(s)}_{i^*+1}} \frac{1}{\tilde{\Delta}^2_{i^*+1} \wedge \tilde{\Delta}^2_{i^*+2}} &= \frac{1}{(\vartheta_{i^*+1}/2 + 1/2 - s)} \frac{1}{\left(-\sqrt{2}\Delta_{i^*}\right)^2 \wedge \Delta^2_{i^*+2}} \\ &\leq \frac{4}{\Delta^2_{i^*} \wedge \Delta^2_{i^*+2}} \leq 4 \max_{i=1,\dots,m} \frac{1}{\Delta^2_i} \leq 4H^\nu_{\mathrm{detect}} \ ,\end{aligned}$$

where one uses that $\max_{i=1,\dots,m} \frac{1}{\Delta^2_i} \leq H^\nu_{\mathrm{detect}}$ because $\vartheta_i \leq 1$ for all $i = 1, \dots, m-1$.

The inequality $H^\nu_{\mathrm{detect}} \leq H^{\tilde{\nu}^{(s)}}_{\mathrm{detect}} \leq 2H^\nu_{\mathrm{detect}}$ follows from the previous inequalities, observing that the case $i = i^*$ gives the lower bound and the other cases give the upper bound. □

Lemma 12. *It holds that*

$$\bar{\mathbb{P}}_\pi(\xi) \leq 2\delta \ .$$

Proof of Lemma 12. Let $e = (1/8 - \eta) > 0$, which is positive because $\eta \leq 1/8$. By definition of $\tilde{l}$ and $\tilde{r}$, one has $\tilde{r} - \tilde{l} = \tilde{\vartheta} + 1/4 > 2\eta + 2e$. In particular, the points $\tilde{l} + e$ and $\tilde{r} - e$ are spaced by at least 2η, so that $\tilde{l} + e + \eta < \frac{1}{2}(\tilde{l} + \tilde{r}) < \tilde{r} - e - \eta$.

Let $\epsilon > 0$ and consider $\bar{\nu}^{(1)}_\epsilon$ as the environment obtained from $\bar{\nu}$ by adding a plateau of height ϵ, and length e on the interval $[\tilde{l}, \tilde{l} + e]$. That is, we consider a change point of jump magnitude $+\epsilon$ at $\tilde{l}$, and another one of jump $-\epsilon$ at $\tilde{l} + e$. Consider also $\bar{\nu}^{(2)}_\epsilon$ as the environment obtained from $\bar{\nu}$ by adding a plateau of height ϵ, and length e on the interval $[\tilde{r} - e, \tilde{r}]$. Note that $\bar{\nu}^{(1)}_\epsilon$ and $\bar{\nu}^{(2)}_\epsilon$ have m valid change points. Moreover, when ϵ goes to zero, the total variation distance between $\bar{\nu}^{(1)}_\epsilon$ and $\bar{\nu}$, and between $\bar{\nu}^{(2)}_\epsilon$ and $\bar{\nu}$ goes to zero[11]. Denote as $\mathbb{P}^{(1)}_\epsilon$ and $\mathbb{P}^{(2)}_\epsilon$ the probability distributions of the observations when the environment is $\bar{\nu}^{(1)}_\epsilon$ and $\bar{\nu}^{(2)}_\epsilon$, respectively, and when the algorithm is π.

Consider the decomposition of the event ξ as

$$\xi = \left\{ \mathcal{T}_\pi \leq \chi, \ \frac{\hat{x}_{i^*} + \hat{x}_{i^*+1}}{2} \leq \frac{\tilde{l} + \tilde{r}}{2} \right\} \sqcup \left\{ \mathcal{T}_\pi \leq \chi, \ \frac{\hat{x}_{i^*} + \hat{x}_{i^*+1}}{2} > \frac{\tilde{l} + \tilde{r}}{2} \right\}$$

Under $\bar{\nu}^{(1)}_\epsilon$, the change point $i^* + 1$ is at position $\tilde{l} + e$. Moreover, it holds that $\tilde{l} + e + \eta < \frac{\tilde{l}+\tilde{r}}{2}$. Then, by the (δ, η, m)-correctness of π, it holds that

$$\mathbb{P}^{(1)}_\epsilon \left(\frac{\hat{x}_{i^*} + \hat{x}_{i^*+1}}{2} > \frac{\tilde{l} + \tilde{r}}{2} \right) \leq \mathbb{P}^{(1)}_\epsilon \left(\hat{x}_{i^*+1} > (\tilde{l} + e) + \eta \right) \leq \delta \ . \tag{46}$$

Similarly, under $\bar{\nu}^{(2)}_\epsilon$, the change point i^* is at position $\tilde{r} - e$. Moreover, it holds that $\tilde{r} - e - \eta \geq \frac{\tilde{l}+\tilde{r}}{2}$. Then, by the (δ, η)-correctness of π, it holds that

$$\mathbb{P}^{(2)}_\epsilon \left(\frac{\hat{x}_{i^*} + \hat{x}_{i^*+1}}{2} \leq \frac{\tilde{l} + \tilde{r}}{2} \right) \leq \mathbb{P}^{(2)}_\epsilon \left(\hat{x}_{i^*} < (\tilde{r} - e) - \eta \right) \leq \delta \ . \tag{47}$$

Overall, we have that, taking the limit $\epsilon \to 0$,

$$\begin{aligned}\bar{\mathbb{P}}_\pi(\xi) &= \lim_{\epsilon \to 0} \left\{ \mathbb{P}^{(1)}_\epsilon \left(\mathcal{T}_\pi \leq \chi, \frac{\hat{x}_{i^*} + \hat{x}_{i^*+1}}{2} > \frac{\tilde{l} + \tilde{r}}{2} \right) + \mathbb{P}^{(2)}_\epsilon \left(\mathcal{T}_\pi \leq \chi, \frac{\hat{x}_{i^*} + \hat{x}_{i^*+1}}{2} \leq \frac{\tilde{l} + \tilde{r}}{2} \right) \right\} \\ &\leq \lim_{\epsilon \to 0} \mathbb{P}^{(1)}_\epsilon \left(\frac{\hat{x}_{i^*} + \hat{x}_{i^*+1}}{2} > \frac{\tilde{l} + \tilde{r}}{2} \right) + \lim_{\epsilon \to 0} \mathbb{P}^{(2)}_\epsilon \left(\frac{\hat{x}_{i^*} + \hat{x}_{i^*+1}}{2} \leq \frac{\tilde{l} + \tilde{r}}{2} \right) \\ &\leq 2\delta \ ,\end{aligned}$$

which concludes the proof of Lemma 12. □

[11] Using the same argument as in Lemma 9

F Relating the continuous and the discrete bandit change point problems

Assume we have an algorithm that is able to localize N change points in a K armed bandit change point problem. This means that if we have K arms with corresponding 1-subGaussian distribution $\tilde{\nu}_k$ and mean $\tilde{\mu}_k$ for $k \in [K]$ and there is $\mathcal{C} = \{1 \leq k_1^* < k_2^* < \cdots < k_m^* < K\}$ such that $\tilde{\mu}_k \neq \tilde{\mu}_{k+1}$ for $k \in \mathcal{C}$, the algorithm is able to identify a subset of $\mathcal{C}$ with cardinality m.

When we are now dealing with a continuous bandit change point problem as introduced in Section 2, we can still tackle it by using such an algorithm.

If we want to localize change points in $[0, 1]$ with precision η, let us first build a discrete bandit, by considering $\tilde{\nu}_k = \nu_{(k-1)\eta}$ for $k = 1, \ldots, \lfloor 1/\eta \rfloor + 1$. If we now localize a change point k^*, it means that $\tilde{\mu}_{k^*} \neq \tilde{\mu}_{k^*+1}$ and therefore $\mu_{(k^*-1)\eta} \neq \mu_{k^*\eta}$. This implies, that in the continuous setting there must be a change point x^* with $(k^* - 1)\eta < x^* \leq k^*\eta$, and using $c = (k^* - 1)\eta$ guarantees that $|x^* - c| \leq \eta$.

While we use this discretization throughout our experiments, we want to remark that conversely it is possible to transform a discrete change point identification problem for K arms to a continuous problem by constructing a respective step function. Here, localizing change points with precision $\eta < 1/2K$ leads to exact change point identification in the discrete case.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The main claims in the abstract and introduction accurately reflect our main contributions from Theorem 1 and Theorem 2. All theorems contain precise statements, with complete assumptions, and all logarithmic factors explicitly available. We also discuss the scope of our claims in Section 6.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitations of our work in Section 6, and open problems (in particular the cases $N < m$ and unknown m) in Appendix B.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: The complete assumptions are clearly stated in the main text. We provide complete proofs in the appendix, and we sketch the main ideas in the main text. Theorem 1 is proved in Appendix D, and Theorem 2 is proved in Appendix E.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We describe five experiments in this paper. For all simulation studies,the Python code is publicly available at `bandit_changepoint`. For all algorithms used, pseudocode is provided or cited in this work. Since we provide the exact experimental setup for each experiment, all results can be fully reproduced.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The code, including all dependencies and instructions on how to reproduce the experiments, is publicly available at `bandit_changepoint`.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: We provide an exact description of the setting, including the choice of all relevant parameters, for every experiment.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Except for Experiment 5, we provide empirical (0.05,0.95)-quantile intervals for the simulated budgets, computed across Monte Carlo runs. These capture the variability due to the random draws in each simulation.

   Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provide a full description of the resources used at the beginning of Section C, including hardware specifications, parallelization, and per-experiment runtimes. No substantial additional compute beyond the reported experiments was required.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have read and understood the code of ethics and adhere to it.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [N/A]

Justification: Our work is mainly theoretical and does not have direct societal impacts.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: We do not release any new models or datasets. Our work is mainly theoretical, and does not pose a risk of misuse.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: We compare our algorithm to methods from prior work, which we cite accordingly. As the referenced works provide pseudocode rather than executable code, our implementations are based solely on the described algorithms and do not reuse any external code assets.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We release the code at `bandit_changepoint` under the MIT license. The repository includes a README with setup instructions and guidance on reproducing all experiments, and documents the differences between the implementation and the pseudocode in the paper.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: This work does not use crowdsourcing or human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: This work does not use crowdsourcing or human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: We did not use for more than editing or formatting purposes.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.