



Brief paper

Change detection using an iterative algorithm with guarantees[☆]Sivaraman Rajaganapathy^{*}, James Melbourne, Murti V. Salapaka

Department of Electrical & Computer Engineering, University of Minnesota, Minneapolis, MN, 55455, USA



ARTICLE INFO

Article history:

Received 10 April 2020

Received in revised form 25 June 2021

Accepted 20 October 2021

Available online 10 December 2021

Keywords:

Step detection algorithm

Iterative improvements

Piece-wise constant signals

ABSTRACT

Multiple domains involve systems with abruptly changing states, that result in signals with signatures that are corrupted by noise and sensor dynamics. In many applications, prior information on the magnitudes and timing distributions are unknown, compounding the difficulty of filtering such signals. Several non-linear filtering techniques are available with varying efficacies, however only a few offer theoretical guarantees. Recent research has led to a heuristic, iterative, non-linear filtering technique that is found to be effective over a range of applications. In this article, we present this iterative algorithm that learns the distribution of the event magnitude and provides an estimate of when an event has occurred. Here, every iteration involves two stages: the first one providing an estimate of the true signal that uses a prior on the distribution of event magnitudes, followed by the second stage where a distribution of the event magnitudes is computed from an estimate of the true signal. It is shown that with every iterate, the performance of the algorithm only improves and convergence of the posterior probability is established. Our comparative tests show that our algorithm provides significant performance improvements, especially for high bandwidth applications where sensor dynamics cannot be neglected. In addition, we provide a python-based implementation of the algorithm and provide a comprehensive comparison with existing methods in practice.

© 2021 Elsevier Ltd. All rights reserved.

1. Introduction

Systems with abrupt changes in their states and parameters arise in a number of diverse domains. The study of such systems thus involves the detection and analysis of signals with abrupt jumps or 'steps'. Examples include detecting DNA copy number variations in bio-informatics (McConnell et al., 2013; Olshen, Venkatraman, Lucito, & Wigler, 2004), failure detection in statistics (Page, 1957; Picard, 1985), detecting edges in images (which are essentially step changes in 2 dimensions) (Canny, 1986; Radke, Andra, Al-Kofahi, & Roysam, 2005), and detecting sudden changes in climate records (Ducré-Robitaille, Vincent, & Boulet, 2003). Many biological systems at the nano-scale are driven by discontinuous processes (Carter, Vershinin, & Gross, 2008; Ha, 2014; Svoboda, Schmidt, Schnapp, & Block, 1993). The study of processes at the molecular scale is crucial for understanding the failure mechanisms of many diseases. For example, molecular motor proteins that transport cargo within cells are

known to take discrete steps on molecular tracks. Defects in such motor proteins are linked with serious diseases such as Huntington's disease, Alzheimer's disease, and breast cancer (Hares et al., 2017; Lucanus & Yip, 2018). Questions such as, how mutations affect step sizes, how multiple motors carry a cargo, whether motors coordinate, what is the impact of the cellular environment on motor performance, are of active research interest (Sweeney & Holzbaur, 2018). Answers to these questions need algorithms that can detect steps with high spatial as well as temporal resolution.

Step signals of interest are rarely noise free. Detecting steps in noisy measurements is a non-trivial problem. Standard linear filters typically work by eliminating noise by removing the high frequency content of the signal. However, the noise and the step signal both have significant power in overlapping ranges in the frequency domain. Consequently linear filters are not well suited for filtering step signals, and often result in the smoothing of the steps or the elimination of short duration steps. The detection problem is further compounded if little or no prior information is available about the underlying generative process such as the distribution of amplitudes and the likely timing of the steps. This is certainly the case when studying new bio-molecules at the single molecule scale. Furthermore, in single molecule investigations, the instruments are operated near the limits of their capabilities, which leads to low signal to noise ratios (SNR) in single molecule data, where the noise magnitudes could be as large as the signals of interest. Similarly, when investigating rapidly changing steps, the

[☆] This project was supported by the NSF CMMI, USA grant awarded to M.S. The material in this paper was partially presented at the 2018 American Control Conference, June 27–29, 2018, Milwaukee, WI, USA. This paper was recommended for publication in revised form by Associate Editor Michele Basseville under the direction of Editor Torsten Söderström.

^{*} Corresponding author.

E-mail addresses: sivrmn@umn.edu (S. Rajaganapathy), james.melbourne@cimat.mx (J. Melbourne), murtis@umn.edu (M.V. Salapaka).

data is further corrupted due to the slow instrument dynamics. Subsequently, a number of nonlinear step detection techniques have arisen across multiple disciplines.

A selection of step detection tools with a focus on molecular biology applications described in Carter and Cross (2005), Hua, Young, Fleming, and Gelles (1997), Kerssemakers et al. (2006), Sadler and Swami (1999) are compared in Carter et al. (2008). The tools in the study were selected from many as they were particularly suited for signals with variable steps sizes and made no assumptions about the generating process. A common feature among most of these algorithms is the presence hyper-parameters that need to be manually tuned for optimal performance. For instance the regularization trade-off weights in Kerssemakers et al. (2006), Rudin, Osher, and Fatemi (1992) and the filter window sizes in Hua et al. (1997), Sadler and Swami (1999) are parameters that need tuning. Tuning these hyper-parameters is not trivial and often requires insight in to the algorithms' mechanics. In contrast, the tools in Aggarwal, Materassi, Davison, Hays, and Salapaka (2012), Mullner, Syed, Selvin, and Sigworth (2010), Syed, Mullner, Selvin, and Sigworth (2010) do not require any hyper-parameter tuning. More recent surveys of offline change detection algorithms are available in Little and Jones (2011), Truong, Oudre, and Vayatis (2020), where the authors classify the algorithms based on the cost function, the penalty function, and the search strategy used in the algorithms.

The algorithm in Aggarwal et al. (2012) is an iterative, dynamic programming based technique that is demonstrated to estimate the stepping probability under low SNRs. Its effectiveness has allowed it to be used in a number of biophysics applications such as the detection of wear in molecular tracks (Dumont, Do, & Hess, 2015), characterizing the unfolding rates of the N2B domain of cardiac protein titin (Pang, Le, & Yan, 2018) and the detection of fluorophore events in the study of nanodot arrays (Cai, 2016).

Inspired by the algorithm in Aggarwal et al. (2012), we provide the step detection algorithm (SDA). The SDA is an iterative algorithm for change detection, where each iteration consists of two stages: a stage where the location and magnitude of steps are determined and the second stage where statistics of the event size is determined from the output of the first stage. The statistics of the second stage is used by the first stage as input in addition to the data. The scheme is entirely data-driven with no prior assumption on event size or timing of events.

The main contributions of this article are: (i) A rigorous non-restrictive probabilistic framework for analyzing the operational steps of the SDA that accommodates sensor dynamics, (ii) A proof of the algorithm's self-improvement property where we show the convergence of the measurement conditioned posterior probability, (iii) An empirical comparison of our implementation of the SDA with the step detection techniques described in Arlot, Celisse, and Harchaoui (2019), Fryzlewicz (2007, 2014), Kerssemakers et al. (2006), Killick, Fearnhead, and Eckley (2012), Mullner et al. (2010), Rudin et al. (1992), (iv) A python-based implementation of the algorithm. In the preliminary conference version (Rajaganapathy, Melbourne, Aggarwal, Shrivastava, & Salapaka, 2018), we presented a theoretical framework for the SDA and sketched out how this can be used to show improvements in estimates with each iteration.

This paper is organized as follows. In Section 2, we formally introduce the step detection problem, the step detection algorithm, and assumptions. In Section 3, we show the iterations of the SDA improve and leading to a convergence of the posterior probability. The comparative empirical performance of the SDA is presented in Section 4 and discussed in Section 5.

2. The step detection problem

2.1. Model assumptions for the detection problem

We will denote stochastic variables by **bold** characters and their realizations by normal characters. A complete sequence will have T elements and will be represented by capital letters, such as $X = \{x_1, x_2, \dots, x_T\}$. A partial sequence will be marked with the starting and ending indices in the sub and super scripts respectively, where $\{x_r, x_2, \dots, x_s\}$ is denoted by X_r^s . The step detection problem statement follows.

Let $X^\dagger = \{x_1^\dagger, x_2^\dagger, \dots, x_T^\dagger\}$ be the true stepping signal generated by a system with its dynamics given by:

$$\mathbf{x}_t^\dagger = \mathbf{x}_{t-1}^\dagger + \mathbf{u}_t^\dagger, \quad (1)$$

with $\mathbf{x}_0^\dagger = 0$, where $\mathbf{u}_t^\dagger$ is a stochastic variable with an unknown distribution $p(\mathbf{u}^\dagger)$, which is independent of time t . In the step detection problem, $\mathbf{u}^\dagger$ represents the magnitude of the steps. Let $Y = \{y_1, y_2, \dots, y_T\}$ be the observations of $X^\dagger$ corrupted by zero mean, Gaussian noise with,

$$\mathbf{y}_t = \mathbf{x}_t^\dagger + \boldsymbol{\eta}_t, \quad (2)$$

where $\{\boldsymbol{\eta}_t\}$ are independent and identically distributed with each $\boldsymbol{\eta}_t \sim \mathcal{N}(0, \sigma^2)$. The step detection problem is to estimate $X^\dagger$ and the density $p(\mathbf{u}^\dagger)$ given the observations Y . We remark that nothing is assumed about the steps $\mathbf{u}_t$, except that it admits a probability distribution function which is time-invariant (thus each $\mathbf{u}_t$, for all $t = 1, \dots, T$ are independent and identically distributed). This very generality is needed for many applications and is also the main reason for the challenges posed.

We approach the problem of estimating the stepping signal by assuming that the probability distribution function $p(u)$ (assumed to be time-invariant) lies in a set of distributions, where each element in the set is determined by a parameter $\Theta \in \mathbb{R}^K$. We denote an element in the class as $p(u|\Theta)$. As the distribution $p(u)$ is time-invariant (i.e. each $\mathbf{u}_t$, for all $t = 1, \dots, T$, being independent and identically distributed), for every Θ , $p(u|\Theta)$ will be assumed to be time-invariant, whereby, $p(U = \{u_1 \dots u_T\} | \Theta) = \prod_{t=1}^T p(u_t | \Theta)$.

We are assuming there is an underlying true distribution $p(\Theta)$ for the parameter $\Theta \in \mathbb{R}^K$.

Fig. 1 shows the Bayesian Network, with vertices and edges, that describe the inter-dependencies of various stochastic variables for the generative model $\mathcal{M}_G$, which synthesizes the steps. Here, as alluded to earlier, $\Theta \in \mathbb{R}^K$ is a random vector for $t \in [0, 1, \dots, T]$. Fig. 1 characterizes the dependencies arising from the structural equations: $\mathbf{x}_{t+1} = \mathbf{x}_t + \mathbf{u}_t$ where $\mathbf{u}_t \sim p(\mathbf{u}|\Theta)$. Thus, Θ and $\mathbf{x}_t$ determine $\mathbf{x}_{t+1}$. Now, $\mathbf{y}_t = \mathbf{x}_t + \boldsymbol{\eta}_t$ and thus $\mathbf{x}_t$ determines $\mathbf{y}_t$. The directed arrows in Fig. 1 go from the right hand side of the equations above to the left hand side variable. In the graphical model, with vertices $\mathcal{V}$ and directed edges $\mathcal{E}$, if $(v_1 \rightarrow v_2)$ exists as an edge, then v_1 is a parent of v_2 and v_2 is a child of v_1 . If two vertices, v_i and v_j have a common child v_k , then v_i and v_j are spouses. If there is a chain $v_i \rightarrow v_{i+1} \rightarrow \dots \rightarrow v_{i+n} \rightarrow v_j$ then i is an ancestor of v_j and v_j is a descendant of v_i .

We now present the concept of a Markov Blanket and a related theorem which will prove to be useful for delineating conditional probabilities.

Definition 1 (Markov Blanket Russell & Norvig, 2016). In a Bayesian Network (such as the one shown in Fig. 1), the Markov Blanket of a node $\mathcal{A}$ is the smallest set of nodes $\mathcal{S}$ in the graph $\mathcal{G}$ such that $\mathcal{A}$ is independent of $\mathcal{G} - \mathcal{S}$ given $\mathcal{S}$. In a Bayesian network, the Markov blanket of $\mathcal{A}$ constitutes the parents, children, and co-parents of the children of $\mathcal{A}$.

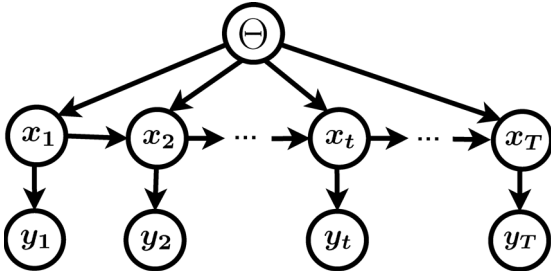


Fig. 1. Underlying generative model assumed for solving the step detection problem.

Theorem 2 (Russell & Norvig, 2016). *The following holds in a Bayesian Network with a vertex set $\mathcal{V}$:*

- (1) Suppose a node v_i has a Markov Blanket M_i . Then $p(v_i|\mathcal{V}\setminus v_i) = p(v_i|M_i)$.
- (2) Suppose $\mathcal{P}_i$ denotes the parents of a node v_i and $\tilde{\mathcal{D}}_i$ denotes all nodes that are not a descendant of v_i . Then $p(v_i|\{\mathcal{P}_i, \tilde{\mathcal{D}}_i\}) = p(v_i|\mathcal{P}_i)$.

Using Bayes rule, we have $p(X, \Theta|Y) = \frac{p(Y|X, \Theta)p(X|\Theta)p(\Theta)}{p(Y)}$. We note that X forms the Markov blanket for Y , thus $p(Y|X, \Theta) = p(Y|X)$. Therefore $p(X, \Theta|Y) = \alpha p(Y|X)p(X|\Theta)p(\Theta)$, where $\alpha = \frac{1}{p(Y)}$. We now expand $p(Y|X)$ and $p(X|\Theta)$ using the Bayes rule as follows: $p(X, \Theta|Y) = \alpha \prod_{t=1}^T p(y_t|Y_1^{t-1}, X)p(x_t|X_1^{t-1}, \Theta)p(\Theta)$.

As x_t forms the Markov blanket for y_t , we have $p(y_t|Y_1^{t-1}, X) = p(y_t|x_t)$. Also $p(x_t|X_1^{t-1}, \Theta) = p(x_t|x_{t-1}, \Theta)$ where we have used the fact from Theorem 2 that x_t is independent of all non-descendants (X_1^{t-2}) given its parents (x_{t-1}, Θ). Thus we get $p(X, \Theta|Y) = \alpha \prod_{t=1}^T p(y_t|x_t)p(x_t|x_{t-1}, \Theta)p(\Theta)$.

2.2. The Step Detection Algorithm

In this section, we now describe the Step Detection Algorithm's (SDA) operational steps in a formal setting. We start by defining a histogram $\mathcal{H}$ and an upper semi-continuous histogram $\bar{\mathcal{H}}$ over a variable u .

Definition 3 (Histogram $\mathcal{H}$). Consider a bounded variable $u \in [u_{\min}, u_{\max}]$. For some finite K , select boundary set $\mathcal{B} = \{b_0, b_1, \dots, b_K\}$ such that

$$b_k - b_{k-1} = \frac{|u_{\max} - u_{\min}|}{K} \text{ for all } k,$$

where $b_0 = u_{\min}$, $b_K = u_{\max}$. Here $r_k := (b_{k-1}, b_k)$, are the open sets between the boundaries in $\mathcal{B}$. We form the bin ranges (also known as bins) $\bar{\mathcal{R}} = \{\bar{r}_1, \bar{r}_2, \dots, \bar{r}_K\}$ as $\bar{r}_k := r_k \cup \{b_{k-1}\}^{\delta_{k-1}} \cup \{b_k\}^{\delta_k}$ for all k , where $\delta_k \in \{0, 1\}$ and $\bar{\delta}_k = 1 - \delta_k$ for all k . Note that whether a boundary element b_k belongs to bin range $\bar{r}_k$ is decided by the variable δ_k . Here, $\mathcal{D} = \{\delta_0, \delta_1, \dots, \delta_K\}$ with $\delta_0 = 0$, and $\delta_K = 1$ are the boundary decision parameters, and

$$\{b\}^\delta := \begin{cases} \text{null set} & \delta = 0 \\ \{b\} & \delta = 1. \end{cases}$$

For a fixed $\mathcal{B}$, with boundary decision parameters $\mathcal{D}$, and bin heights $\Theta = \{\theta_1, \theta_2, \dots, \theta_K\}$, $\theta_k \in \mathbb{R}$, the histogram $\mathcal{H}$ over a bounded variable u is defined as

$$\mathcal{H}_{\Theta, \mathcal{D}}(u) = \sum_{k=1}^K \theta_k \mathbb{I}(u \in \bar{r}_k).$$

Definition 4 (Upper Semi-continuous Histogram $\bar{\mathcal{H}}$, see Fig. 2). Given a height set Θ , the corresponding histogram $\mathcal{H}_{\Theta, \mathcal{D}}(u) =$

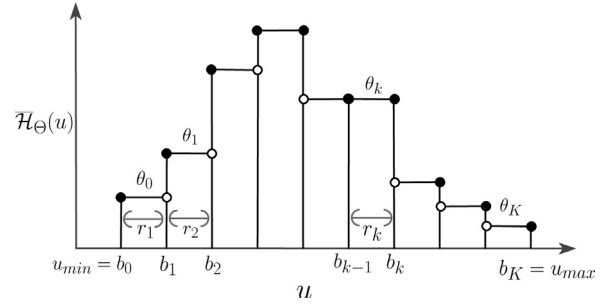


Fig. 2. An upper semi-continuous histogram $\bar{\mathcal{H}}_{\Theta}(u)$.

$\sum_{k=1}^K \theta_k \mathbb{I}(u \in \bar{r}_k)$ is upper semi-continuous if

$$\delta_k = \begin{cases} 1 & \theta_k \geq \theta_{k+1} \\ 0 & \theta_k < \theta_{k+1} \end{cases} \text{ for all } k \neq 0, K,$$

with $\delta_0 = 0$, and $\delta_K = 1$. We denote upper semi-continuous histograms as:

$$\bar{\mathcal{H}}_{\Theta}(u) = \sum_{k=1}^K \theta_k \mathbb{I}(u \in \bar{r}_k).$$

Note that a upper semi-continuous histogram $\bar{\mathcal{H}}$, defined over $u \in [u_{\min}, u_{\max}]$, is parameterized by the bin heights $\Theta = \{\theta_1, \theta_2, \dots, \theta_K\}$, and that definition 4 agrees with the usual, more general, definition of upper semi-continuity. For simplicity from this point on, 'histogram' will be used to mean the upper semi-continuous histogram $\bar{\mathcal{H}}$ unless otherwise stated. The SDA approximates the underlying distribution $\gamma(\cdot)$ of u by a histogram $\bar{\mathcal{H}}$ as described in Definition 4. Note: In the implementation of the SDA, the number of bins of the histogram K is selected based on the resolution r needed in estimating the step magnitudes. That is, $K = \frac{|u_{\max} - u_{\min}|}{r}$.

The SDA is composed of two primary stages, namely the minimization stage (M-Stage) followed by the evaluation stage (E-Stage) that at the n th iteration generates an estimate $[\hat{X}^{(n)}, \Theta^{(n)}]$. Let $n \in \{1, 2, \dots, N\}$ represent the iteration count. A description of the two stages is as follows:

M-Stage: For a given $\Theta^{(n-1)}$, minimize the objective function: $J(X, \Theta^{(n-1)}) := \sum_{t=1}^T (y_t - x_t)^2 + W_t(X, \Theta^{(n-1)})$, with respect to X . Let

$$\hat{X}^{(n)} := \arg \min_X J(X, \Theta^{(n-1)}). \quad (3)$$

Here, $W_t(X, \Theta^{(n-1)})$ is a regularization term defined as:

$$W_t(X, \Theta^{(n-1)}) = \begin{cases} 9\sigma^2 \bar{\delta}(x_t - x_{t-1}) & n = 1 \\ -2\sigma^2 \log[p(u_t|\Theta^{(n-1)})] & n > 1 \end{cases} \quad (4)$$

$$\text{with } \bar{\delta}(u) = \begin{cases} 0 & u = 0 \\ 1 & \text{otherwise.} \end{cases}$$

Note that, $p(u_t|\Theta^{(n-1)})$ is the probability density function of taking a step u_t evaluated for the step size u_t , with $u_t = x_t - x_{t-1}$ in iteration n , given $\Theta^{(n-1)}$, the previous estimate of parameters of the step size distribution. For the M-Stage, $\Theta^{(n-1)}$ is assumed to be known.

E-Stage: Consider a given $\hat{X}^{(n)}$. From the given $\hat{X}^{(n)}$, let $u_t^{(n)} := \hat{x}_{t+1}^{(n)} - \hat{x}_t^{(n)}$. Evaluate $\Theta^{(n)} = \{\theta_1^{(n)}, \theta_2^{(n)}, \dots, \theta_K^{(n)}\}$ and select $\mathcal{D}^{(n)} = \{\delta_0^{(n)}, \delta_1^{(n)}, \dots, \delta_K^{(n)}\}$ with $\delta_0^{(n)} = 0$, and $\delta_K^{(n)} = 1$ using

$$\theta_k^{(n)} = \frac{S_k^{(n)}}{T} \text{ for all } k = 1, \dots, T, \quad (5)$$

such that $\Theta^{(n)}$ and $\mathcal{D}^{(n)}$ form an upper semi-continuous histogram.

Here $s_k^{(n)}$ is the number of steps, $u_t^{(n)}$, in the fit $\hat{X}^{(n)}$ contained within the bin $\bar{r}_k$. Thus $s_k^{(n)} = |\{t : u_t^{(n)} \in \bar{r}_k\}|_1$ where $|\Gamma|_1$ is the cardinality of the set Γ . Here T is the total number of observations of steps $u_t^{(n)}$ in the estimate $\hat{X}^{(n)}$, thus $T = \sum_{k=1}^K s_k^{(n)}$ (note that in the computation of T and s_k , $\{t : u_t = 0\}$ are also counted as being steps of zero amplitude).

An alternative to fixing the total number of iterations to N is to evaluate the M and E-Stages $|\hat{X}^n - \hat{X}^{n-1}| \leq \varepsilon_X$ and $|\Theta^n - \Theta^{n-1}| \leq \varepsilon_\Theta$ for some given tolerances ε_X and ε_Θ . Assuming N is known, the SDA is summarized in Algorithm 1.

Algorithm 1 Step Detection Algorithm

Input: Time series measurements $Y = \{y_1, y_2, \dots, y_T\}$.

Output: Step fit $\hat{X}^{(N)} = \{\hat{x}_1^{(N)}, \hat{x}_2^{(N)}, \dots, \hat{x}_T^{(N)}\}$ and Histogram bin parameters $\Theta^{(N)} = \{\theta_1^{(N)}, \theta_2^{(N)}, \dots, \theta_K^{(N)}\}$.

- 1: Set initial penalty using (4)
 - 2: Compute step fit $\hat{X}^{(1)}$ using (3)
 - 3: **for all** $n \in \{2, 3, \dots, N\}$ **do**
 - 4: **E-Stage:** Compute $\Theta^{(n)}$ using (5)
 - 5: **M-Stage:** Compute step fit $\hat{X}^{(n)}$ using (3)
 - 6: **end for**
-

3. Convergence of the Step Detection Algorithm

In this section, we analyze the convergence properties of the Step Detection Algorithm.

3.1. Increments in the joint conditional probability

We now show that each iteration consisting of the M-Stage and E-Stage of the SDA improves upon the previous estimates whose joint conditional probability density $p(\hat{X}, \Theta|Y)$ evaluated at $\hat{X}^{(n)}$, $\Theta^{(n)}$ improves with each iterate. The SDA uses the knowledge gained from the previous estimates in the M-Stage, in the form of a previously estimated distribution $\mathcal{H}_{\Theta^{(n-1)}}(u)$ of the step sizes.

In the E-stage, the step size distribution parameters $\Theta^{(n)}$ are computed from $\hat{X}^{(n)}$ alone, without any direct use of previous estimates $\Theta^{(i)}$ and $\hat{X}^{(i)}$, for $i = 1, \dots, n-1$. In this manner, the estimate $\Theta^{(n)}$ is kept 'unbiased', wherein all $\Theta \in \mathbb{R}^K$ that form a valid histogram, are given equal preference for being selected as $\Theta^{(n)}$. On the contrary, utilizing information of the previous estimates in computing $\Theta^{(n)}$, would imply some histograms are favored over others.

Assumption 1. At the E-Stage of the SDA, the computation of $\Theta^{(n)}$ is unbiased. That is, the step size histogram with unknown bin heights $\Theta = \{\theta_1, \dots, \theta_K\}$, is uniformly distributed over a measurable set of finite volume.

For example, consider the case where the distribution Θ is represented by a histogram with K bins $\{\theta_1, \theta_2, \dots, \theta_K\}$. Then, for any integer K , $\Omega = \{\Theta : \sum_{i=1}^K \theta_i = 1, \theta_i \geq 0\}$, is a compact set of a K dimensional affine subspace $\mathbb{R}^K$, and thus can be endowed with a uniform probability measure. That is, we assume for any two histograms Θ' and Θ'' , $p(\Theta') = p(\Theta'') = \frac{1}{\mathcal{A}_K}$. Here, $\mathcal{A}_K$ is the area of the $K-1$ dimensional hyper plane $\mathcal{S}^{K-1}(\lambda = 1)$, where $\mathcal{S}^{K-1}(\lambda) := \{\Theta \in \mathbb{R}^K | \sum_{k=1}^K \theta_k = \lambda, \theta_k \geq 0 \text{ for all } k\}$. The area $\mathcal{A}_K$ is given by $\mathcal{A}_K = \frac{\sqrt{K}}{(K-1)!}$.

Lemma 5. Maximizing the probability $p(X, \Theta|Y)$ with respect to X under the assumed Bayesian Network of Fig. 1 is equivalent to minimizing $J(X, \Theta)$ with respect to X , where,

$$J(X, \Theta) = \sum_{t=1}^T (y_t - x_t)^2 - 2\sigma^2 \log [p(\mathbf{u}_t | X_1^{t-1}, \Theta)].$$

That is, if $\hat{X}^* = \arg \max_X p(X, \Theta|Y)$ then $\hat{X}^* = \arg \min_X J(X, \Theta)$.

Proof. We observe that by Bayes rule,

$$p(X, \Theta|Y) = \frac{p(Y|X, \Theta)p(X|\Theta)p(\Theta)}{p(Y)}.$$

Since Y is independent of Θ given X (X forms the Markov Blanket (Russell & Norvig, 2016) for Y , see model in Fig. 1), $p(Y|X, \Theta) = p(Y|X)$, and since $p(\Theta)$ and $p(Y)$ do not depend on X . Thus, $\hat{X}^* = \arg \max_X p(X, \Theta|Y) = \arg \max_X p(Y|X)p(X|\Theta)$.

By repeated application of Bayes Rule, we get $\hat{X}^* = \arg \max_X \prod_{t=1}^T p(y_t | Y_1^{t-1}, X) p(x_t | X_1^{t-1}, \Theta)$. Using $\mathbf{y}_t = \mathbf{x}_t + \boldsymbol{\eta}_t$, we observe that $\mathbf{y}_t \sim \mathcal{N}(x_t, \sigma^2)$. We then have,

$$\begin{aligned} \hat{X}^* &= \arg \max_X \prod_{t=1}^T \frac{\exp \left\{ -\frac{(y_t - x_t)^2}{2\sigma^2} \right\}}{\sqrt{2\pi\sigma^2}} p(x_t | X_1^{t-1}, \Theta) \\ &= \arg \min_X \sum_{t=1}^T \frac{(y_t - x_t)^2}{2\sigma^2} - \log p(\mathbf{u}_t = u_t | X_1^{t-1}, \Theta), \end{aligned}$$

with, $u_t = x_t - x_{t-1}$. Thus,

$$\hat{X}^* = \arg \min_X \sum_{t=1}^T (y_t - x_t)^2 - 2\sigma^2 \log p(\mathbf{u}_t | X_1^{t-1}, \Theta).$$

This completes the proof. $\square$

Lemma 6. For Θ uniformly distributed with $p(X, \Theta|Y)$ and $p(U|\Theta)$ governed by the Bayesian Network of Fig. 1,

$$\Theta^* = \arg \max_{\Theta} p(X, \Theta|Y) = \arg \max_{\Theta} p(U|\Theta).$$

Proof. Using the definition of conditional probability, we note $\Theta^* = \arg \max_{\Theta} p(X, \Theta|Y) = \arg \max_{\Theta} p(\Theta|X, Y)$. However, according to our model structure, Θ is independent of Y given X (X forms the Markov Blanket (Russell & Norvig, 2016) for Θ). Thus, $\Theta^* = \arg \max_{\Theta} p(\Theta|X) = \arg \max_{\Theta} p(X|\Theta) \times p(\Theta)$.

Since $p(\Theta)$ is assumed to be uniform, $\Theta^* = \arg \max_{\Theta} p(X|\Theta)$.

We use Bayes rule to re-write $p(X|\Theta)$ as

$$\prod_{t=1}^T p(\mathbf{x}_t = x_t | X_1^{t-1}, \Theta) = \prod_{t=1}^T p(\mathbf{u}_t = u_t | \Theta).$$

Thus, $p(X|\Theta) = p(U|\Theta)$. Therefore,

$$\arg \max_{\Theta} p(X, \Theta|Y) = \arg \max_{\Theta} p(U|\Theta).$$

This proves the assertion. $\square$

Now we consolidate the results to establish that SDA improves the posterior probability with every iterate.

Theorem 7. For Θ uniformly distributed, let the observations Y be corrupted by zero mean Gaussian noise with known variance σ^2 . That is, $\mathbf{y}_t = \mathbf{x}_t + \boldsymbol{\eta}_t$, with $\boldsymbol{\eta}_t \sim \mathcal{N}(0, \sigma^2)$ for all t . Then, given an estimate $[\hat{X}^{(n-1)}, \Theta^{(n-1)}]$, the step detection algorithm's iteration produces an improved estimate $[\hat{X}^{(n)}, \Theta^{(n)}]$ whereby

$$p(\hat{X}^{(n)}, \Theta^{(n)}|Y) \geq p(\hat{X}^{(n-1)}, \Theta^{(n-1)}|Y).$$

Proof. From Lemma 5, we note that the maximization of $p(X, \Theta|Y)$ with respect to X is equivalent to a minimization of $J(X, \Theta)$ with respect to X .

Now, in the **M-Stage** of SDA, we have $\Theta = \Theta^{(n-1)}$ and minimize $J(X, \Theta^{(n-1)})$ with respect to X to give $\hat{X}^{(n)}$. This is equivalent to maximizing $p(X, \Theta^{(n-1)}|Y)$ with respect to X . Therefore

$$p(X^{(n)}, \Theta^{(n-1)}|Y) \geq p(X, \Theta^{(n-1)}|Y) \text{ for all } X. \quad (6)$$

In the **E-Stage** of SDA, we compute the bin heights of the upper semi-continuous histogram as described in (5), which represents the step size distribution, using the estimate $\hat{X}^{(n)}$ available at stage n . In Lemma 8, we will show that this binning process is equivalent to maximizing $p(\hat{X}^{(n)}|\Theta)$ with respect to Θ to give $\Theta^{(n)}$. From Lemma 6, we observe that

$$\Theta^{(n)} = \arg \max_{\Theta} p(U^{(n)}|\Theta) = \arg \max_{\Theta} p(\hat{X}^{(n)}, \Theta|Y).$$

Therefore,

$$p(\hat{X}^{(n)}, \Theta^{(n)}|Y) \geq p(\hat{X}^{(n)}, \Theta|Y) \quad \forall \Theta. \quad (7)$$

Consequently, from (6) and (7), we see that each iteration results in

$$\begin{aligned} p(\hat{X}^{(n)}, \Theta^{(n)}|Y) &\geq p(\hat{X}^{(n)}, \Theta^{(n-1)}|Y) \\ &\geq p(\hat{X}^{(n-1)}, \Theta^{(n-1)}|Y). \end{aligned}$$

This completes the proof. $\square$

Remark. In the implementation, we compute a histogram of the step sizes U , with bin heights $\Theta = \{\theta_1, \theta_2, \dots, \theta_K\}$. In Lemma 8 we show that building such a histogram is equivalent to maximizing $p(U^{(n)}|\Theta)$ with respect to Θ , to give $\Theta^{(n)}$.

We briefly compare the similar features of the two algorithms as follows. From Lemma 5, Theorem 7, and (4), we observe that the SDA performs coordinate ascent on $p(X, \Theta|Y)$. With the assumption that $p(\Theta)$ is uniform (i.e. a constant), and the observation that both $p(\Theta)$ and $p(Y)$ are independent of the values X and Θ , we have

$$\begin{aligned} \arg \max_{X \text{ or } \Theta} p(X, \Theta|Y) &= \arg \max_{X \text{ or } \Theta} p(X, \Theta|Y) p(Y) \\ &= \arg \max_{X \text{ or } \Theta} p(X, Y|\Theta) p(\Theta) \\ &= \arg \max_{X \text{ or } \Theta} p(X, Y|\Theta). \end{aligned}$$

Thus, the SDA performs a coordinate ascent on $p(X, Y|\Theta)$.

The EM algorithm maximizes the function $Q(f(X), \Theta)$, where $Q(f(X), \Theta) := \sum_{X \in \mathcal{X}} f(X) \left\{ \ln \frac{p(X, Y|\Theta)}{f(X)} \right\}$, and $f(X)$ is a probability density function of X . Specifically, the EM algorithm achieves this via coordinate ascent in two steps. In the E-step, the EM algorithm fixes $\Theta = \Theta^{n-1}$ and maximizes $Q(f(X), \Theta)$ along $f(X)$ by the assignment $f(X) = p(X|Y, \Theta^{n-1})$. In the M-step, the EM algorithm fixes $f(X) = p(X|Y, \Theta^{n-1})$, and maximizes $Q(f(X), \Theta)$ along Θ (Bilmes et al., 1998). Thus, the SDA's coordinate maximization of $p(X^{n-1}, Y|\Theta)$ along Θ can be thought of as the coordinate maximization of $Q(f(X), \Theta)$ along Θ by the EM algorithm, under the substitution $f(X) = \delta(X - X^{n-1})$.

3.2. Binning produces the histogram with the best fit

In Lemma 8, we prove our earlier claim that building a histogram of the steps U in the estimate $\hat{X}$ in the E-Stage is equivalent to maximizing the probability $p(U|\Theta)$ with respect to the histogram bin heights Θ and thus provides the “best fit”.

Lemma 8. Let $U = \{u_1, u_2, \dots, u_T\}$ be the observations of a stationary process, that is, the probability distribution of u_t is independent of t . Let $\bar{\mathcal{H}}$ be an upper semi-continuous histogram over u as described in Definition 4, approximating the distribution of U . Let the bin heights be $\Theta = \{\theta_1, \theta_2, \dots, \theta_K\}$. Let s_k be the number of observations u_t contained within the bin $\bar{r}_k$, that is $s_k(U, \Theta) = |\{t : u_t \in \bar{r}_k\}|$. Then, the local maxima of $\Theta \mapsto p(U|\Theta)$ satisfy $\theta_k = \frac{s_k(U, \Theta)}{T}$.

Proof. Let

$$\Theta^* := \arg \max_{\Theta} p(U|\Theta).$$

Here each u_t are independent and identically distributed. We then approximate the density $p(u_t|\Theta)$ by the histogram $\bar{\mathcal{H}}_{\Theta}(u_t)$, thus

$$\Theta^* = \arg \max_{\Theta} \prod_{t=1}^T p(u_t|\Theta) = \arg \max_{\Theta} \prod_{t=1}^T \bar{\mathcal{H}}_{\Theta}(u_t)$$

$$\begin{aligned} &= \arg \max_{\Theta} \prod_{t=1}^T \sum_{k=1}^K \theta_k \mathbb{I}(u_t \in \bar{r}_k) = \arg \max_{\Theta} \prod_{k=1}^K \theta_k^{s_k} \\ &= \arg \max_{\Theta} \sum_{k=1}^K s_k \log [\theta_k]. \end{aligned}$$

Therefore

$$\Theta^* = \arg \max_{\Theta} p(U|\Theta) = \arg \max_{\Theta} \sum_{k=1}^K s_k \log [\theta_k]. \quad (8)$$

We now optimize $f(\Theta) = \sum_{k=1}^K s_k \log [\theta_k]$ with respect to bin heights Θ , subject to the equality constraint $g(\Theta) = \sum_{k=1}^K \theta_k - \frac{K}{u_{\max} - u_{\min}} = 0$. We form the Lagrangian to solve for this constrained optimum.

Note that, despite $s_k(U, \Theta)$ being a discontinuous function with respect to Θ , $f(\Theta)$ is continuous with respect to Θ .

Further, it can be shown, that the points at which $f(\Theta)$ is non-differentiable cannot be its maximizers and that $\nabla_{\Theta} f(\Theta)$ exists near any maximizer Θ^* . With that, we form the Lagrangian: $\mathcal{Q}(\Theta, \lambda) := f(\Theta) - \lambda g(\Theta)$.

The necessary condition for optimality gives

$\nabla \mathcal{Q}(\Theta^*, \lambda^*) = \bar{\mathbf{0}}$. This is equivalent to

$$\left[\frac{s_1}{\theta_1^*} - \lambda^*, \quad \dots, \quad \frac{s_K}{\theta_K^*} - \lambda^*, \quad \sum_{k=1}^K \theta_k^* - \frac{K}{u_{\max} - u_{\min}} \right] = \bar{\mathbf{0}}.$$

This implies that $T = \sum_{k=1}^K s_k = \sum_{k=1}^K \lambda^* \theta_k^* = \frac{K \lambda^*}{u_{\max} - u_{\min}}$. We then have $\theta_k^* = \frac{K s_k}{T(u_{\max} - u_{\min})}$. We further note that

$$\nabla^2 \mathcal{Q}(\Theta^*, \lambda^*) = \begin{bmatrix} -\frac{s_1}{(\theta_1^*)^2} & 0 & 0 & 0 \\ 0 & -\frac{s_2}{(\theta_2^*)^2} & 0 & 0 \\ & & \ddots & \vdots \\ 0 & 0 & \dots & -\frac{s_K}{(\theta_K^*)^2} & 0 \\ 0 & 0 & & 0 & 0 \end{bmatrix}.$$

Here we observe that $\nabla^2 \mathcal{Q}(\Theta^*, \lambda^*)$ is negative semi-definite, which shows that Θ^* is indeed a local maximizer of $p(U|\Theta)$, proving the assertion. $\square$

3.3. SDA proof of convergence

From (6) and (7) we note that the M-Stage and the E-Stage are maximizing $p(X, \Theta|Y)$ with respect to X and Θ respectively. That is $X^{(n)} = \arg \max_X p(X, \Theta^{(n-1)}|Y)$, and $\Theta^{(n)} = \arg \max_{\Theta} p(X^{(n)}, \Theta|Y)$.

The E and M stages can therefore be thought of as the stages of cyclic block coordinate descent (coordinate ascent, but we use ‘descent’ for convenience and uniformity with literature), with $p(X, \Theta|Y)$ as the objective function being maximized. We now find an expression for the objective function when the density $p(u_t|\Theta)$ is governed by a histogram $\bar{\mathcal{H}}_{\Theta}(u_t)$. As observed in Lemma 5, using Bayes rule, $p(X, \Theta|Y) = \frac{p(Y|X, \Theta)p(X|\Theta)p(\Theta)}{p(Y)}$. Here $p(Y)$ is not dependent on either X or Θ , and $p(\Theta)$ is assumed to be uniform. Further Y is independent of Θ given X in the model $\mathcal{M}_{\mathcal{G}}$. Therefore,

$$\arg \max_{X \text{ or } \Theta} p(X, \Theta|Y) = \arg \max_{X \text{ or } \Theta} p(Y|X)p(X|\Theta).$$

From (2),

$$p(Y|X) = \frac{1}{\sqrt{2\pi}\sigma^2} \exp\left(-\frac{(y_t - x_t)^2}{2\sigma^2}\right).$$

As seen in Lemma 6, $p(X|\Theta) = p(U|\Theta)$. Now u_t are independent and identically distributed as per our model $\mathcal{M}_{\mathcal{G}}$. So $p(U|\Theta) = \prod_{t=1}^T p(u_t|\Theta)$. Using the histogram $\bar{\mathcal{H}}_{\Theta}(u)$ for approximating the density $p(u|\Theta)$ as seen in Lemma 8 we get

$p(X|\Theta) = \prod_{k=1}^K \theta_k^{s_k}$. Thus, $\zeta := \arg \max_{X \text{ or } \Theta} p(Y|X)p(X|\Theta)$ can be shown to be

$$\begin{aligned} \zeta &= \arg \max_{X \text{ or } \Theta} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(\sum_{t=1}^T -\frac{(y_t - x_t)^2}{2\sigma^2}\right) \prod_{k=1}^K \theta_k^{s_k} \\ &= \arg \max_{X \text{ or } \Theta} \sum_{t=1}^T -(y_t - x_t)^2 + \sum_{k=1}^K 2\sigma^2 s_k \log(\theta_k) \\ &= \arg \max_{X \text{ or } \Theta} \mathcal{J}(X, \Theta). \end{aligned}$$

Theorem 9. For a finite length bounded observation sequence Y , the sequence

$$\{p(\hat{X}^{(1)}, \Theta^{(1)}|Y), p(\hat{X}^{(2)}, \Theta^{(2)}|Y), \dots, p(\hat{X}^{(n)}, \Theta^{(n)}|Y), \dots\}$$

generated by the step detection algorithm converges.

Proof. We observe from theorem 7 that

$p(\hat{X}^{(n)}, \Theta^{(n)}|Y) \geq p(\hat{X}^{(n-1)}, \Theta^{(n-1)}|Y)$ for all n . Thus by monotone convergence, the sequence $\{p(\hat{X}^{(n)}, \Theta^{(n)}|Y)\}$ converges to a bounded limit.

Further $\{(\hat{X}, \Theta)\}$ are chosen from a compact set: Θ is chosen from a simplex of finite and non-zero volume, and the candidates for $\hat{X}$ are selected from a compact set, indeed $x_t \in [x_0 + tu_{\min}, x_0 + tu_{\max}]$ for all $t = 1, \dots, T$. Note that $p(\hat{X}^{(n)}, \Theta^{(n)}|Y)$ is a probability density and hence non-negative for every n . For a finite number of bins K , with $u_{\max} \neq u_{\min}$, $\sup(\bar{\mathcal{H}}_{\Theta}(U^{(n)})) = \frac{K}{u_{\max} - u_{\min}}$. Therefore,

$$0 \leq p(\hat{X}^{(n)}, \Theta^{(n)}|Y) \leq \frac{K}{u_{\max} - u_{\min}} \text{ for every } n.$$

This implies that $\{(\hat{X}^{(n)}, \Theta^{(n)})\}$ contain a convergent subsequence $\{(\hat{X}^{(m)}, \Theta^{(m)})\}$. $\square$

Proposition 10. For every $\Theta \in S^{K-1}(\lambda)$, the set $\{X^* \in \arg \max_X \mathcal{J}(X, \Theta)\}$ is non-empty. Further, a necessary condition for local maximization is that for $t \geq 1$, $x_t^* \neq y_t$ either

$$x_t^* - x_{t-1}^* \in \mathcal{B} \text{ or } x_{t+1}^* - x_t^* \in \mathcal{B},$$

must hold.

Proof. For fixed Θ , $\mathcal{J}$ is an upper semi-continuous function in X taking at least one finite value on a compact domain $\{X : x_t - x_{t-1} \in [u_{\min}, u_{\max}]\}$. Choosing a sequence X_n such that $\lim_n \mathcal{J}(X_n, \Theta) = \sup_X \mathcal{J}(X, \Theta)$, and a subsequential limit X^* by compactness. By upper semicontinuity $\sup_X \mathcal{J}(X, \Theta) = \lim_n \mathcal{J}(X_n, \Theta) \leq \mathcal{J}(X^*, \Theta)$. Thus X^* is a maximizer.

If X is such that there exists t such that $x_t \neq y_t$ and neither $x_t - x_{t-1}$ nor $x_{t+1} - x_t$ belong to $\mathcal{B}$, then there exists a $\varepsilon > 0$ such that $s_k(X + \delta e_t, \Theta) = s_k(X, \Theta)$ for all k when $|\delta| < \varepsilon$ where e_t denotes the t th canonical basis vector. Since the s_k are unchanged under small perturbations of e_t , $\mathcal{J}(X + \delta e_t, \Theta) - \mathcal{J}(X, \Theta) = -\|X + \delta e_t - Y\|_2^2 + \|X - Y\|_2^2$. Differentiating in δ ,

$$\frac{d}{d\delta} \mathcal{J}(X + \delta e_t, \Theta) \Big|_{\delta=0} = -2(x_t - y_t) \neq 0.$$

Thus X is not a maximizer. $\square$

Lemma 11. Every $X = \{x_1, x_2, \dots, x_T\}$ admits an upper semi-continuous histogram with $\mathcal{J}(X, \bar{\mathcal{H}}_{\Theta}(U)) \geq \mathcal{J}(X, \mathcal{H}_{\Theta, \mathcal{D}}(U))$, where

$$\mathcal{J}(X, \mathcal{H}_{\Theta, \mathcal{D}}(U)) := \sum_{t=1}^T -(y_t - x_t)^2 + 2\sigma^2 \sum_{k=1}^K \log \mathcal{H}_{\Theta, \mathcal{D}}(u_t).$$

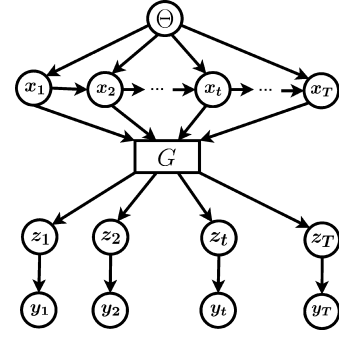


Fig. 3. Underlying generative model $\mathcal{M}_G^S$ assumed for solving the step detection problem, with the addition of sensor dynamics represented by G .

Proof. We observe that

$$\begin{aligned} \Delta_{\mathcal{J}} &:= \mathcal{J}(X, \bar{\mathcal{H}}_{\Theta}(U)) - \mathcal{J}(X, \mathcal{H}_{\Theta, \mathcal{D}}(U)) \\ &= \sum_{t=1}^T \left[-(y_t - x_t)^2 + 2\sigma^2 \sum_{k=1}^K \log \bar{\mathcal{H}}_{\Theta}(u_t) \right] \\ &\quad - \sum_{t=1}^T \left[-(y_t - x_t)^2 + 2\sigma^2 \sum_{k=1}^K \log \mathcal{H}_{\Theta, \mathcal{D}}(u_t) \right] \\ &= 2\sigma^2 \sum_{t=1}^T \sum_{k=1}^K (\log \bar{\mathcal{H}}_{\Theta}(u_t) - \log \mathcal{H}_{\Theta, \mathcal{D}}(u_t)). \end{aligned}$$

Let $u_t \notin \mathcal{B}$ for all $t = \alpha_1, \alpha_2, \dots, \alpha_A$ and $u_t \in \mathcal{B}$ for all $t = \beta_1, \beta_2, \dots, \beta_B$. The descriptions of the histogram in Definition 3 and the upper semi-continuous histogram in Definition 4 imply that $\bar{\mathcal{H}}_{\Theta}(u_t) = \mathcal{H}_{\Theta, \mathcal{D}}(u_t)$ for all $t \in \{\alpha_1, \alpha_2, \dots, \alpha_A\}$. Further for all $b_k \in \mathcal{B}$

$$\bar{\mathcal{H}}_{\Theta}(b_k) = \max\{\theta_k, \theta_{k+1}\},$$

while

$$\mathcal{H}_{\Theta, \mathcal{D}}(b_k) = \begin{cases} \theta_k & \delta_k = 1 \\ \theta_{k+1} & \delta_k = 0, \end{cases}$$

where $\delta_k \in \mathcal{D}$, the boundary selection variable. Therefore, for any choice of $\mathcal{D}$

$$\bar{\mathcal{H}}_{\Theta}(u_t) \geq \mathcal{H}_{\Theta, \mathcal{D}}(u_t) \text{ for all } t \in \{\beta_1, \beta_2, \dots, \beta_B\}.$$

This gives us

$$\begin{aligned} \Delta_{\mathcal{J}} &= 2\sigma^2 \sum_{t=\alpha_1}^{\alpha_A} \sum_{k=1}^K (\log \bar{\mathcal{H}}_{\Theta}(u_t) - \log \mathcal{H}_{\Theta, \mathcal{D}}(u_t)) \\ &\quad + 2\sigma^2 \sum_{t=\beta_1}^{\beta_B} \sum_{k=1}^K (\log \bar{\mathcal{H}}_{\Theta}(u_t) - \log \mathcal{H}_{\Theta, \mathcal{D}}(u_t)) \\ &= 2\sigma^2 \sum_{t=\beta_1}^{\beta_B} \sum_{k=1}^K (\log \bar{\mathcal{H}}_{\Theta}(u_t) - \log \mathcal{H}_{\Theta, \mathcal{D}}(u_t)) \geq 0. \quad \square \end{aligned}$$

Note: $\mathcal{J}(X, \bar{\mathcal{H}}_{\Theta}(U)) = \mathcal{L}(X, \Theta) = \sum_{t=1}^T -(y_t - x_t)^2 + 2\sigma^2 \sum_{k=1}^K s_k \log[\theta_k]$.

3.4. SDA formulation with sensor dynamics

Measurements of stepping signals are often done with sensors whose dynamics have time constants which are of the same time scale as the stepping rates, and thus cannot be ignored. In such settings, accounting for the sensor dynamics improves the estimates.

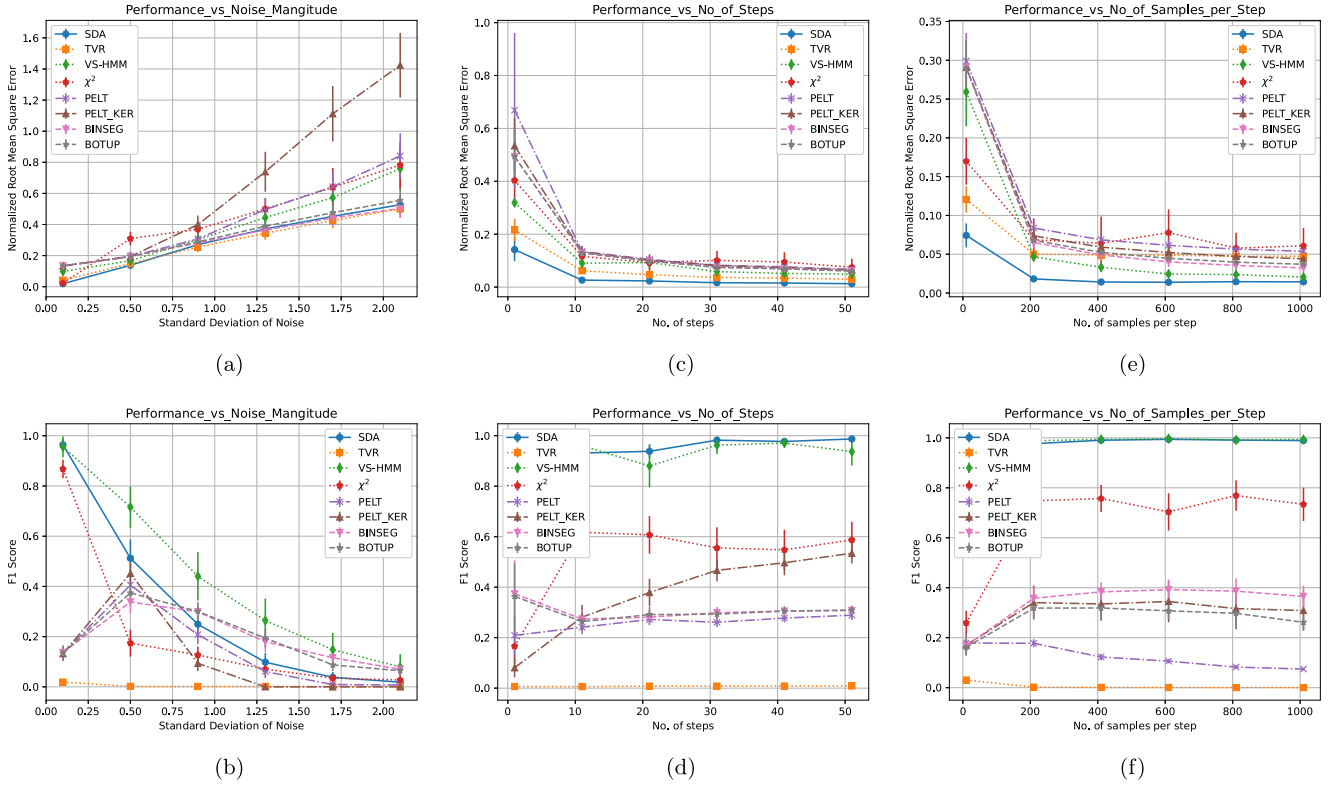


Fig. 4. Comparative performance of the SDA with: (a) & (b) increasing noise magnitude; (c) & (d) increasing number of steps available in the signal; (e) & (f) increasing number of samples per step in the signal.

Let $g(\tau)$ be the discrete time impulse response and $G(z)$ be the Z-domain transfer function of the sensor. Let z_t be the output of the sensor at time t . Then $z_t = \sum_{l=-\infty}^{\infty} g(l)x(t-l)$, where we have further identified $x_t \equiv x(t)$. The reformulated SDA is summarized in Algorithm 2 and is given as follows:

M-stage: Only the M-Stage is modified, wherein the objective function $J_G(X, \Theta^{(n-1)})$ is minimized with respect to X to give $\hat{X}^{(n)}$, that is

$$\hat{X}^{(n)} := \arg \min_X J_G(X, \Theta^{(n-1)}), \quad (9)$$

with $J_G(X, \Theta^{(n-1)}) := \sum_{t=1}^T (y_t - z_t)^2 + W_t(X, \Theta^{(n-1)})$, where $z_t = \sum_{l=-\infty}^{\infty} g(l)x(t-l)$, and

$$W_t(X, \Theta^{(n-1)}) = \begin{cases} 9\sigma^2 \delta(x_t - x_{t-1}) & n = 1 \\ -2\sigma^2 \log[p(u_t | \Theta^{(n-1)})] & n > 1. \end{cases} \quad (10)$$

E-Stage: This step remains unchanged from the standard SDA as described in (5). This modified generative model $\mathcal{M}_G^s$ which now includes the sensor dynamics is represented by the Bayesian Network shown in Fig. 3. Under this reformulation, it can be shown that Theorem 7 holds, i.e. given an estimate $[X^{(n-1)}, \Theta^{(n-1)}]$, the modified SDA provides an improved estimate $[X^{(n)}, \Theta^{(n)}]$ such that $p(X^{(n)}, \Theta^{(n)} | Y) \geq p(X^{(n-1)}, \Theta^{(n-1)} | Y)$.

4. Results and comparisons

We now compare the performance of the SDA with the following step detection methods: Total Variation Regularization (TVR) (Boyd & Vandenberghe, 2004), Hidden Markov Models (VS-HMM) (Mullner et al., 2010; Syed et al., 2010), the χ^2 based method developed by Kerssemakers (χ^2) (Kerssemakers et al., 2006), Linearly Penalized Segmentation (PELT) based search method described in Killick et al. (2012), Gaussian Kernel

Algorithm 2 The SDA with Sensor Dynamics

Input: Time series measurements $Y = \{y_1, y_2, \dots, y_T\}$ and sensor dynamic model $G(z)$.

Output: Step fit $\hat{X}^{(N)} = \{\hat{x}_1^{(N)}, \hat{x}_2^{(N)}, \dots, \hat{x}_T^{(N)}\}$ and Histogram bin parameters $\Theta^{(N)} = \{\theta_1^{(N)}, \theta_2^{(N)}, \dots, \theta_K^{(N)}\}$.

- 1: Set initial penalty using (10)
- 2: Compute step fit $\hat{X}^{(1)}$ using (9)
- 3: **for all** $n \in \{2, 3, \dots, N\}$ **do**
- 4: **E-Stage:** Compute $\Theta^{(n)}$ using (5)
- 5: **M-Stage:** Compute step fit $\hat{X}^{(n)}$ using (9)
- 6: **end for**

(PELT_KER) (Arlot et al., 2019), Binary-Segmentation (BINSEG) (Fryzlewicz, 2014), and Bottom-up segmentation (BOTUP) (Fryzlewicz, 2007) methods.

4.1. Performance metrics

The normalized root mean square error (NRMSE) is used for measuring the difference between the ground truth $X^\dagger$ and the estimates $\hat{X}$, where the NRMSE is defined by $e_{NRMSE} := \frac{1}{C}$

$$\sqrt{\frac{1}{T} \sum_{t=1}^T (x_t^\dagger - \hat{x}_t)^2}, \text{ with normalization constant } C :=$$

$$\sqrt{\frac{1}{T} \sum_{t=1}^T (x_t^\dagger - \bar{x})^2}, \text{ and } \bar{x} = \frac{1}{T} \sum_{t=1}^T x_t^\dagger. \text{ The performance of the}$$

algorithms is also compared when the step detection algorithms are used as binary classifiers, wherein they predict the presence or absence of a step at any given time. Specifically, when the magnitude of an estimated step is predicted to be within $\pm 10\%$ of the magnitude of the true step, and the location of such a step is within ± 5 time steps of the true location, it is considered

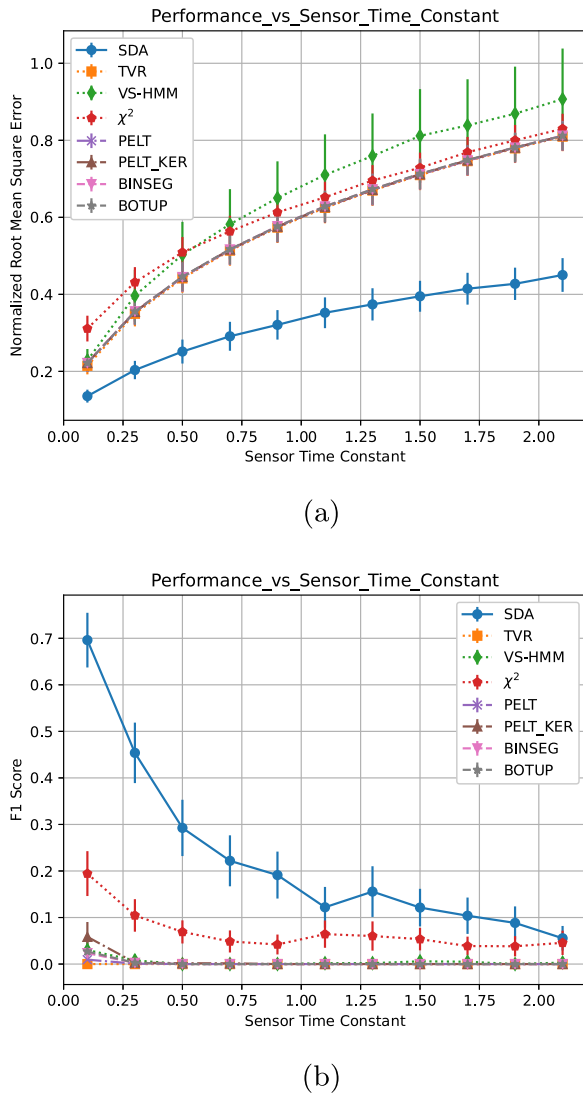


Fig. 5. Comparative performance of the SDA with sensor dynamics. (a) The SDA has a significantly lower NRMSE as it takes into account the sensor dynamics. (b) As expected the SDA has a larger F1 score compared to the other algorithms which do not account for sensor dynamics.

a true positive (TP). Any estimated steps of incorrect magnitude or location with respect to the previously described criterion are considered to be false positives (FP). When the algorithms fail to estimate a step within the location tolerance band of a true step, a false negative (FN) is assigned.

We have used the F1 score (Goodfellow, Bengio, & Courville, 2016) as a classification performance metric, which is defined as $F1 = \frac{2 \cdot (PPV)(TPR)}{PPV + TPR}$, where the positive predictive value is $PPV := \frac{TP}{TP + FP}$, and the true positive rate $TPR := \frac{TP}{TP + FN}$.

4.2. Empirical results & discussion

All the algorithms were tested on identical batches of randomly generated sample stepping signals to ensure a fair comparison.

The plots in Figs. 4 and 5 show the mean of the NRMSE and F1 scores from multiple ($N=50$) runs. The error bars indicate 95% confidence intervals evaluated assuming the samples of the performance metric follow the Student's t distribution (Winkler, 1993).

In Fig. 4(a) & (b), the performance of the algorithms is measured as the standard deviation of the noise is increased from 0.1 to 2.0. It can be seen that the SDA balances both the NRMSE and F1 score performance measures, following closely the best algorithms with respect to each metric. Fig. 4(c) & (d) compare the performance as the number of steps in a test signal are increased from 1 to 50. A larger number of steps in the signal improves the performance of all the algorithms. Fig. 4(e) & (f) compare performance of the algorithms as the average number of samples per step is raised from 10 to 1000. Note that in all the tests without sensor dynamics, the TVR along with the SDA provides the best NRMSE. Whereas, the VS-HMM, together with the SDA provides the best F1 scores.

In Fig. 5, the algorithms are tested on step signals filtered by first order sensor transfer function given by $G(s) = \frac{1}{\tau s + 1}$. The sensor time constant τ is increased from 0.1 to 2 s, where the signals are sampled at 50 Hz, and the average duration of a step is 1 s. It is seen that the SDA which can take into account the dynamics, has a 30%–40% reduction in NRMSE compared to other techniques.

5. Conclusions

The need for detecting steps in noisy signals is frequently encountered in a number of fields. We have provided a Step Detection algorithm, where we provide a theoretical foundation for analysis and show that each iteration of the algorithm necessarily improves the posterior probability of estimates given the observations. Further, the iterations result in a converging sequence of the posterior probability. We also provide a new technical toolbox for the analysis of iterations that uses histograms to approximate probability density functions. The improvements in estimates hold in the presence sensor dynamics, a situation often encountered, for example, when studying the physics of biological systems at the molecular scale. Comparative empirical analysis supports the theoretical description and the SDA's efficacy.

Python toolbox

The python implementation of the algorithm is available here: <https://github.umn.edu/rajag038/SDA.git>.

Acknowledgments

We would like to thank Dr. Jacob W.J. Kerssemakers and Prof. Marileen Dogterom for sharing the MATLAB implementation for their χ^2 based step fitting algorithm for benchmarking tests.

References

- Aggarwal, T., Materassi, D., Davison, R., Hays, T., & Salapaka, M. (2012). Detection of steps in single molecule data. *Cellular and Molecular Bioengineering*, 5(1), 14–31.
- Arlot, S., Celisse, A., & Harchaoui, Z. (2019). A kernel multiple change-point algorithm via model selection. (p. 56).
- Bilmes, J. A., et al. (1998). A gentle tutorial of the EM algorithm and its application to parameter estimation for Gaussian mixture and hidden Markov models, vol. 4, no. 510. (p. 126). International Computer Science Institute.
- Boyd, S. P., & Vandenberghe, L. (2004). *Convex optimization*. Cambridge, UK ; New York: Cambridge University Press.
- Cai, H. (2016). *Biomimetic nanoarchitectures for the study of T cell activation with single-molecule control*. Columbia University.
- Canny, J. (1986). A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6), 679–698.
- Carter, N. J., & Cross, R. (2005). Mechanics of the kinesin step. *Nature*, 435(7040), 308.
- Carter, B. C., Vershinin, M., & Gross, S. P. (2008). A comparison of step-detection methods: how well can you do? *Biophysical Journal*, 94(1), 306–319.

- Ducré-Robitaille, J.-F., Vincent, L. A., & Boulet, G. (2003). Comparison of techniques for detection of discontinuities in temperature series. *International Journal of Climatology: A Journal of the Royal Meteorological Society*, 23(9), 1087–1101.
- Dumont, E. L., Do, C., & Hess, H. (2015). Molecular wear of microtubules propelled by surface-adhered kinesins. *Nature Nanotechnology*, 10(2), 166–169.
- Fryzlewicz, P. (2007). Unbalanced haar technique for nonparametric function estimation. *Journal of the American Statistical Association*, 102(480), 1318–1327.
- Fryzlewicz, P. (2014). Wild binary segmentation for multiple change-point detection. *The Annals of Statistics*, 42(6).
- Goodfellow, I., Bengio, Y., & Courville (2016). *Deep learning*. MIT Press.
- Ha, T. (2014). Single-molecule methods leap ahead. *Nature Methods*, 11(10), 1015–1018.
- Hares, K., Miners, J. S., Cook, A. J., Rice, C., Scolding, N., Love, S., et al. (2017). Overexpression of kinesin superfamily motor proteins in alzheimers disease. In I. Gozes (Ed.), *Journal of Alzheimer's Disease*, 60(4), 1511–1524.
- Hua, W., Young, E. C., Fleming, M. L., & Gelles, J. (1997). Coupling of kinesin steps to ATP hydrolysis. *Nature*, 388(6640), 390.
- Kerssemakers, J. W., Munteanu, E. L., Laan, L., Noetzel, T. L., Janson, M. E., & Dogterom, M. (2006). Assembly dynamics of microtubules at molecular resolution. *Nature*, 442(7103), 709.
- Killick, R., Fearnhead, P., & Eckley, I. A. (2012). Optimal detection of change-points with a linear computational cost. *Journal of the American Statistical Association*, 107(500), 1590–1598.
- Little, M. A., & Jones, N. S. (2011). Generalized methods and solvers for noise removal from piecewise constant signals. I. background theory. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 467(2135), 3088–3114.
- Lucanus, A. J., & Yip, G. W. (2018). Kinesin superfamily: roles in breast cancer, patient prognosis and therapeutics. *Oncogene*, 37(7), 833–838.
- McConnell, M. J., Lindberg, M. R., Brennand, K. J., Piper, J. C., Voet, T., Cowing-Zitron, C., et al. (2013). Mosaic copy number variation in human neurons. *Science*, 342(6158), 632–637.
- Mullner, F. E., Syed, S., Selvin, P. R., & Sigworth, F. J. (2010). Improved hidden Markov models for molecular motors, part 1: basic theory. *Biophysical Journal*, 99(11), 3684–3695.
- Olshen, A. B., Venkatraman, E. S., Lucito, R., & Wigler, M. (2004). Circular binary segmentation for the analysis of array-based DNA copy number data. *Biostatistics*, 5(4), 557–572.
- Page, E. S. (1957). On problems in which a change in a parameter occurs at an unknown point. *Biometrika*, 44(1/2), 248–252.
- Pang, S. M., Le, S., & Yan, J. (2018). Mechanical responses of the mechanosensitive unstructured domains in cardiac titin. *Biology of the Cell*, 110(3), 65–76.
- Picard, D. (1985). Testing and estimating change-points in time series. *Advances in Applied Probability*, 17(4), 841–867.
- Radke, R., Andra, S., Al-Kofahi, O., & Roysam, B. (2005). Image change detection algorithms: a systematic survey. *IEEE Transactions on Image Processing*, 14(3), 294–307.
- Rajaganapathy, S., Melbourne, J., Aggarwal, T., Shrivastava, R., & Salapaka, M. V. (2018). Learning and estimation of single molecule behavior. In: *2018 Annual American control conference* (pp. 5125–5130). ISSN: 2378-5861.
- Rudin, L. I., Osher, S., & Fatemi, E. (1992). Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1–4), 259–268.
- Russell, S. J., & Norvig, P. (2016). *Artificial intelligence: A modern approach*. Malaysia; Pearson Education Limited.
- Sadler, B. M., & Swami, A. (1999). Analysis of multiscale products for step detection and estimation. *IEEE Transactions on Information Theory*, 45(3), 1043–1051.
- Svoboda, K., Schmidt, C. F., Schnapp, B. J., & Block, S. M. (1993). Direct observation of kinesin stepping by optical trapping interferometry. *Nature*, 365(6448), 721–727.
- Sweeney, H. L., & Holzbaur, E. L. (2018). Motor proteins. *Cold Spring Harbor Perspectives in Biology*, 10(5), a021931.
- Syed, S., Mullner, F. E., Selvin, P. R., & Sigworth, F. J. (2010). Improved hidden Markov models for molecular motors, part 2: extensions and application to experimental data. *Biophysical Journal*, 99(11), 3696–3703.
- Truong, C., Oudre, L., & Vayatis, N. (2020). Selective review of offline change point detection methods. *Signal Processing*, 167, Article 107299.
- Winkler, J. R. (1993). *Numerical recipes in C: The art of scientific computing*, second edition. *Endeavour*, 17(4), 201.



Sivaraman Rajaganapathy receive his bachelor's in Electrical Engineering from the University of Mumbai, in 2011 and his Master's in Systems & Control Engineering from the Indian Institute of Technology, Bombay, in 2013. Subsequently he worked on automation of testing and validation of semiconductor devices and sensors at Cypress Semiconductor Corp. He is currently pursuing his Ph.D. at the University of Minnesota, Twin-Cities, with a major in Electrical Engineering and a minor in Computer Science. His thesis research is at the intersection of statistical mechanics, biophysics, and mathematical modeling aimed towards studying proteins at the single molecule scale.



James Melbourne received his Master's degree in Mathematics with emphasis in Finance from the University of Kansas in 2009, and his Ph.D. in Mathematics at the University of Minnesota in 2015. He was a postdoctoral researcher in the Mathematics department at University of Delaware from 2015 to 2017, and in the Computer and Electrical Engineering department of the University of Minnesota from 2017 to 2020, before accepting a position in the Probability and Statistics department of the CIMAT mathematical research institute in Guanajuato, Mexico, where he is currently an assistant professor. His research interests include convexity theory, probability, convex geometry, information theory, and their applications.



Murli V. Salapaka received the B.Tech. degree in Mechanical Engineering from the Indian Institute of Technology, Madras, in 1991 and the M.S. and Ph.D. degrees in Mechanical Engineering from the University of California at Santa Barbara, in 1993 and 1997, respectively. He was a faculty member in the Electrical and Computer Engineering Department, Iowa State University, Ames, from 1997 to 2007. Currently, he is the Director of Graduate Studies and the Vincentine Hermes Luh Chair Professor in the Electrical and Computer Engineering Department, University of Minnesota, Minneapolis. His research interests include control and network science, nanoscience and single molecule physics. Dr. Salapaka received the 1997 National Science Foundation CAREER Award and is an IEEE fellow.