# BLOCK-WISE MAP INFERENCE FOR DETERMINANTAL POINT PROCESSES WITH APPLICATION TO CHANGE-POINT DETECTION

*Martin J. Zhang*

Dept. of EE, Stanford University, CA, USA, jinye@stanford.edu

*Zhijian Ou*

Dept. of EE, Tsinghua University, Beijing, China, ozj@tsinghua.edu.cn

## ABSTRACT

Most studies of change-point detection (CPD) focus on developing similarity metrics that quantify how likely a time-point is to be a change point. After that, the process of selecting true change points among those high-score candidates is less well-studied. This paper proposes a new CPD method that uses determinantal point processes to model the process of change-point selection. Specifically, this work explores the particular kernel structure arose in such modelling, the almost block diagonal. It shows that the *maximum a posteriori* task, requiring at least $O(N^{2.4})$ in general, can be achieved using $O(N)$ under such structure. The resulting algorithms, BwDPP-MAP and BwDppCpd, are empirically validated through simulation and five real-world data experiments.

***Index Terms*—** Change-point detection, determinantal point processes, MAP inference

## 1. INTRODUCTION

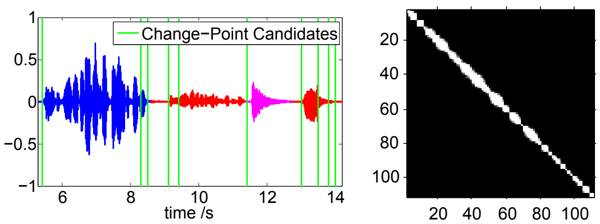
The determinantal point processes (DPPs) are elegant probabilistic models for subset selection problems where both quality and diversity are considered. Formally, a DPP, specified by an L-ensemble (positive semi-definite) kernel $\mathbf{L} \in \mathbb{R}^{N\times N}$, defines a probability measure $\mathcal{P}$ over all subsets of a point set $\mathcal{Y} = \{1, \cdots, N\}$, where the probability mass function is $\mathcal{P}_{\mathbf{L}}(Y) \propto \det(\mathbf{L}_Y)$, $\forall\, Y \subset \mathcal{Y}$. The DPP kernels are usually built through quality-diversity decomposition, i.e.

$$\mathbf{L} = \mathrm{diag}(\mathbf{q})\mathbf{S}\,\mathrm{diag}(\mathbf{q}), \tag{1}$$

where $\mathrm{diag}(\mathbf{q})$ is a diagonal matrix formed by the quality vector $\mathbf{q}$, assigning a quality score to each item in $\mathcal{Y}$, and $\mathbf{S}$ is called the similarity matrix, quantifying the similarity between every pair of items. DPPs constructed in such way assign a higher probability to subsets whose elements are of higher quality and lower similarity [2]. In other words, they favour both quality and diversity.

The DPP *maximum a posteriori* (MAP) problem, i.e. finding the subset with the highest probability, is NP-hard [3]. A few approximate inference methods are proposed, including greedy methods for optimizing the submodular function $\log\det(L_Y)$ [4], optimization via continuous relaxation [5], and minimum Bayes risk decoding [2]. The first has a computational complexity $O(N^{2.4})$, the second $O(N^3)$, and the third $O(RT^2N\log N/\epsilon)$[1], all super linear w.r.t. $N$.

In the first part of the paper, we show that for DPPs with an almost block diagonal kernel, which we call BwDPPs (block-wise DPPs), it is possible to achieve linear computational complexity w.r.t. $N$ for MAP inference. The algorithm that achieves this, named BwDPP-MAP, calls existing DPP-MAP algorithms on carefully tailored sub-blocks of the full kernel to solve the global optimization problem approximately, with a minor sacrifice of the inference accuracy. The sub-inference scale does not grow with $N$ and the number of sub-inferences grows proportional to $N$, giving the linear dependence on $N$. In the second part of the paper, we apply BwDPP-MAP in the change-point detection problem (CPD), which aims at detecting abrupt changes in time-series data.




(a) (b)

**Fig. 1**. (a) A 10-sec part from a 2-min speech. Vertical lines show change-point candidates and colors indicate different states (speaker or noise change). (b) The corresponding BwDPP kernel for the entire 2-min speech. The white denotes non-zero entries while the black indicates zero.

In CPD, the methods are roughly classified as Bayesian or frequentist. Bayesian approaches [7, 8, 9] focus on estimating the posterior distribution of change-point locations given the time-series data, where the computational cost is challenging, especially for real-world tasks. Frequentist approaches usually consist of two steps. First, calculate a metric score for each time point, quantifying if a change happens there based on its past and future segments; second, select change points based on the metric scores. The first step is well-studied, e.g. the generalized likelihood ratio [10], the Bayesian information criterion (BIC) [11], the Kullback Leibler divergence [12]. One can also refer to [13, 14, 15, 16, 17, 18] for more results. However, the second step, change-point selection, is relative lack of study. Some immature methods include selecting local peaks above a threshold [15], discarding the lower one if two peaks are close [19], or requiring the metric differences between change-points and their neighbouring valleys above a threshold [12].

Based on BwDPP-MAP, we developed a new two-step CPD method, BwDppCpd. In the first step, it takes advantage of existing well-studied metrics to select a preliminary set of change-point candidates. In the second step, the change-point selection process is achieved by constructing a DPP kernel by the quality-diversity decomposition and performing MAP inference by BwDPP-MAP. Specifically, each change-point candidate has its quality of being a

This work is supported by NSFC grant 61473168 and partly from M.J. Zhang's undergraduate thesis in Tsinghua University [1], advised by Z. Ou.

[1] $R \sim 1000$ is the number of Monte Carlo simulations; the rest is the best complexity for DPP sampling, achieved by [6], where $T$ is the size of result generated by the DPP sampling algorithm, and $\epsilon$ is the approximation error.

change-point, and locations of true change-points should be diverse since states do not change rapidly. These are addressed by the quality vector and similarity matrix respectively. Moreover, only nearby time points are similar to each other, making the DPP kernel almost block diagonal, i.e. BwDPP. Such behaviour is illustrated in Fig. 1.

In the rest of the paper, we first introduce BwDPP-MAP and BwDppCpd in Section 2 and then present evaluation experiments on five real-world datasets in Section 3. Proofs are appended in the end.

## 2. METHODOLOGY

In this paper, we focus on almost block diagonal DPP kernels. Formally, a $\gamma$-almost block diagonal matrix has the form

$$\mathbf{L} \triangleq \begin{bmatrix} \mathbf{L}_1 & \mathbf{A}_1 & & \cdots & \mathbf{0} \\ \mathbf{A}_1^T & \mathbf{L}_2 & \mathbf{A}_2 & & \\ & \ddots & \ddots & \ddots & \vdots \\ & & \mathbf{A}_{m-2}^T & \mathbf{L}_{m-1} & \mathbf{A}_{m-1} \\ \mathbf{0} & \cdots & & \mathbf{A}_{m-1}^T & \mathbf{L}_m \end{bmatrix}, \quad (2)$$

where diagonal components $\mathbf{L}_i \in \mathbb{R}^{l_i \times l_i}$ are dense matrices, and off-diagonal components $\mathbf{A}_i \in \mathbb{R}^{l_i \times l_{i+1}}$ have non-zero entries only at bottom left, whose size does not exceed $\gamma \times \gamma$. A DPP kernel will have such structure when items are only similar to their neighbours, as mentioned above.

As a side note, almost block diagonal matrices have two properties: 1. block-tridiagonal, 2. sparse off-diagonal blocks. As shown below, the first gives linear computational complexity in the determinant calculation, similar as in previous works for general block tridiagonal matrices [20, 21]. The second helps to reduce the inference error but has nothing to do with the determinant calculation.

For the matrix $\mathbf{L}$, let $\mathcal{Y}$ be its index set and let $\mathcal{Y}_i$ be the set of indices corresponding to $\mathbf{L}_i$, for $i = 1, \dots, m$. For any $C_i, C_j \subseteq \mathcal{Y}$, by $\mathbf{L}_{C_i,C_j}$ we mean the sub-matrix with rows and columns specified by $C_i$ and $C_j$ respectively, and $\mathbf{L}_{C_i,C_i}$ is abbreviated as $\mathbf{L}_{C_i}$. Finally, we note that $\mathbf{L} \succeq 0$ means that $\mathbf{L}$ is positive semi-definite.

### 2.1. BwDPP-MAP: Fast MAP Inference for BwDPPs

Let $\mathbf{L}$ be any almost block diagonal kernel defined in (2). Let $C \subseteq \mathcal{Y}$ be the hypothesized subset to be selected from $\mathbf{L}$ and let $C_i \subseteq \mathcal{Y}_i$ be that from $\mathbf{L}_i, \forall i \in \{1, \cdots, m\}$. We note that $C_i = C \cap \mathcal{Y}_i$. Assume $\mathbf{L}_{C_i}$ is invertible, $\forall i \in \{1, \cdots, m\}$. By defining $\tilde{\mathbf{L}}_{C_i}$ recursively as $\tilde{\mathbf{L}}_{C_i} \triangleq$

$$\begin{cases} \mathbf{L}_{C_i} & i = 1, \\ \mathbf{L}_{C_i} - \mathbf{L}_{C_{i-1},C_i}^T \tilde{\mathbf{L}}_{C_{i-1}}^{-1} \mathbf{L}_{C_{i-1},C_i} & i = 2, \cdots, m \end{cases},$$

one could rewrite the MAP objective function: $\det(\mathbf{L}_C)$

$$= \det(\mathbf{L}_{C_1}) \det(\mathbf{L}_{\cup_{i=2}^m C_2} - \mathbf{L}_{C_1,\cup_{i=2}^m C_i}^T \mathbf{L}_{C_1}^{-1} \mathbf{L}_{C_1,\cup_{i=2}^m C_i})$$
$$= \det(\tilde{\mathbf{L}}_{C_1}) \det(\begin{bmatrix} \tilde{\mathbf{L}}_{C_2} & [\mathbf{L}_{C_2,C_3}\ \mathbf{0}] \\ [\mathbf{L}_{C_2,C_3}\ \mathbf{0}]^T & \mathbf{L}_{\cup_{i=3}^m C_i} \end{bmatrix}),$$

where $\mathbf{0}$ is the zero matrix of appropriate size. The second equation holds because $\mathbf{L}_{C_1,C_i} = \mathbf{0}$ for $i \geq 3$, noting that L is almost diagonal. Continuing the recursion to $m$, we have

$$\det(\mathbf{L}_C) = \cdots = \textstyle\prod_{i=1}^m \det(\tilde{\mathbf{L}}_{C_i}),$$

which converts the MAP inference problem to

$$\operatorname*{argmax}_{C \in \mathcal{Y}} \det(\mathbf{L}_C) = \operatorname*{argmax}_{C_1 \in \mathcal{Y}_1, \cdots, C_m \in \mathcal{Y}_m} \textstyle\prod_{i=1}^m \det(\tilde{\mathbf{L}}_{C_i}).$$

**Table 1**. BwDPP-MAP Algorithm

| |
|---|
| **Input:** $\mathbf{L}$ as defined in (2); Any DPP-MAP algorithm. |
| **Output:** Subset of items $\hat{C}$. |
| **For:** $i = 1, \cdots, m$ |
| Compute $\tilde{\mathbf{L}}_{\mathcal{Y}_i}$ via (2.1); |
| Perform sub-inference over $C_i$ using DPP-MAP via |
| $\hat{C}_i = \mathrm{argmax}_{C_i \in \mathcal{Y}_i; C_j = \hat{C}_j, j=1,\cdots,i-1} \det((\tilde{\mathbf{L}}_{\mathcal{Y}_i})_{C_i})$; |
| **Return:** $\hat{C} = \bigcup_{i=1}^m \hat{C}_i$. |

Instead of maximizing $\det(\mathbf{L}_C)$, we maximize $\det(\tilde{\mathbf{L}}_{C_i})$ separately for each $i$, and report the merged answer. By doing so we implicitly assume that

$$\operatorname*{argmax}_{C_i \in \mathcal{Y}_i} \textstyle\prod_{i=1}^m \det(\tilde{\mathbf{L}}_{C_i}) \approx \cup_{i=1}^m \operatorname*{argmax}_{C_i \in \mathcal{Y}_i} \det(\tilde{\mathbf{L}}_{C_i}). \quad (3)$$

This is reasonable because the almost block diagonal structure ensures that $\tilde{\mathbf{L}}_{C_i}$ has a weak correlation with the subsets other than $C_i$, which further indicates the approximation error is small, validating such method. The resulting sub-inference method, BwDPP-MAP, is described in Table 1. For notation, $\mathrm{argmax}_{C_i; C_j = \hat{C}_j, j=1,\cdots,i-1}$ denotes optimizing over $C_i$ with the value of $C_j$ fixed as $\hat{C}_j$ for $j = 1, \cdots, i-1$, and the sub-kernel $\tilde{\mathbf{L}}_{\mathcal{Y}_i}$ is given similarly as $\tilde{\mathbf{L}}_{C_i}$, namely $\tilde{\mathbf{L}}_{\mathcal{Y}_i} \triangleq$

$$\begin{cases} \mathbf{L}_i & i = 1, \\ \mathbf{L}_i - \mathbf{L}_{C_{i-1},\mathcal{Y}_i}^T \tilde{\mathbf{L}}_{C_{i-1}}^{-1} \mathbf{L}_{C_{i-1},\mathcal{Y}_i} & i = 2, \cdots, m \end{cases}$$

One may notice that $(\tilde{\mathbf{L}}_{\mathcal{Y}_i})_{C_i}$ is equivalent to $\tilde{\mathbf{L}}_{C_i}$.

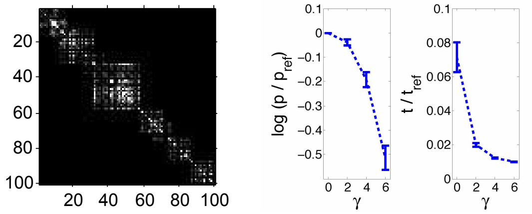
**Remark 1** *Any DPP-MAP algorithm can be plugged in for the BwDPP sub-inference, because DPP-MAP algorithms take positive semi-definite matrices as input, and it can be shown that $\tilde{\mathbf{L}}_{\mathcal{Y}_i} \succeq 0$, for $i = 1, \cdots, m$ (the proof is postponed to the end of the paper). Hence, BwDPP-MAP is a universal booster for any DPP-MAP algorithm. If some more advanced DPP-MAP algorithm comes out, BwDPP-MAP can directly use them for a performance boost.*

**Remark 2** *Fixing the DPP-MAP algorithm, compared to directly applying it on the entire kernel, the computational cost saving by BwDPP-MAP can be significant. Let the kernel size $N$ grows and the sub-kernel size roughly remains some constant c, which is the case for CPD, where the sub-block sizes are only decided by how a time-point related to its neighbours. Suppose the DPP-MAP has an $O(N^\alpha)$ computational complexity. For BwDPP-MAP, each sub-inference takes $O(c^\alpha)$ time, and there are $N/c$ sub-inferences, yielding a $O(Nc^{\alpha-1}) = O(N)$ complexity. In this example, BwDPP-MAP boosts the speed from $O(N^\alpha)$ to $O(N)$, where $\alpha \geq 2.4$.*

**Remark 3** *In practice, first we need to specify $\gamma$ and then partition the kernel accordingly [2]. The different choice of $\gamma$ represents a speed-error tradeoff: on one hand, as $\gamma$ increases, the sub-kernel size will decrease, reducing the computational complexity of sub-inference, and further the overall complexity. On the other hand, increasing $\gamma$ will make neighbouring sub-kernels more connected to each other, which deteriorates the assumption (3) and introduces more error.*

*We provide an empirical example in Fig. 2, where (1) 1000 independent simulations of kernels of size 500; (2) sub-kernel size:*

[2] Concretely, the partition method in this paper is to (1) identify as many as possible non-overlapping dense diagonal sub-matrices; (2) merge adjacent sub-matrices if their off-diagonal non-zero area size exceeds $\gamma \times \gamma$.




(a) (b)

**Fig. 2**. (a) The top-left $100 \times 100$ entries of a $500 \times 500$ synthetic kernel. (b) The log-probability ratio $\log(p/p_{\text{ref}})$ and runtime ratio $t/t_{\text{ref}}$, with 1000 repetition and 99.7% error bar, of running BwDPP-MAP with different $\gamma$-partition. The reference performance, is produced by running greedy-MAP on the entire kernel.

**Table 2**. Greedy-MAP Algorithm

**Input:** $\mathbf{L}$; **Output:** $\hat{C}$.

**Initialization:** Set $\hat{C} \leftarrow \emptyset, U \leftarrow \mathcal{Y}$;
**While** $U$ is not empty;

$i^* \leftarrow \operatorname{argmax}_{i \in U} L_{ii}$; $\hat{C} \leftarrow \hat{C} \cup \{i^*\}$;

Compute $\mathbf{L}^* = \left( \left[ (\mathbf{L} + \mathbf{I}_{\bar{\hat{C}}})^{-1} \right]_{\bar{\hat{C}}} \right)^{-1} - \mathbf{I}$;

$\mathbf{L} \leftarrow \mathbf{L}^*$; $U \leftarrow \{i | i \notin \hat{C}, \mathbf{L}_{ii} > 1\}$;

**Return:** $\hat{C}$.

$10 - 30$, *non-zero off-diagonal areas:* $\{0, 2, 4, 6\}$, *randomly chosen; (3) values of non-zero entries are given as the Gram product of random Gaussian vectors; (4) greedy-MAP (Table 2) [5] is used for BwDPP-MAP sub-inference; Fig. 2 (a) shows an example of such synthetic kernels. The BwDPP-MAP aims to approximate the inference result produced by the reference, with much faster speed. Fig. 2 (b) validates such thought. As* $\gamma$ *increases, the runtime drops fast while the inference accuracy degrades very slow.*

### 2.2. BwDppCpd: BwDPP-based Change-Point Detection

Let $\mathbf{x}_1, \cdots, \mathbf{x}_T \in \mathbb{R}^D$ be the time-series observations, and let $\mathbf{x}_{\tau:t}$ denote the observation segment from $\tau$ to $t$. We use $\mathbf{X}_1$, $\mathbf{X}_2$ to denote different segments for simplicity. A dissimilarity metric is denoted by $d : (\mathbf{X}_1, \mathbf{X}_2) \mapsto \mathbb{R}$, which measures the dissimilarity between segments[3]. Our CPD method, BwDppCpd, is a two-step method described as below.

**Step 1: locating change-point candidates.** Given a dissimilarity metric $d$, a pair of adjacent length-$w$ windows slides along the timeline to calculate the dissimilarity score of each time point, i.e. $d(\mathbf{x}_{t-w+1:t}, \mathbf{x}_{t+1:t+w})$. Then, locations of local peaks above score mean, $t_1, \cdots, t_N$, are selected as change-point candidates $\mathcal{Y} = \{1, \cdots, N\}$.

**Step 2: change-point selection via BwDPP.** Construct the kernel $\mathbf{L}$ as $\mathbf{L} = \operatorname{diag}(\mathbf{q}) * \mathbf{S} * \operatorname{diag}(\mathbf{q})$, where $\mathbf{q}$ is the quality vector with element $q_i = d(\mathbf{x}_{t_{i-1}:t_i}, \mathbf{x}_{t_i:t_{i+1}})$, and $\mathbf{S}$ is the similarity matrix with $S_{ij} \triangleq \exp(-(t_i - t_j)^2/\sigma^2)$, where $\sigma$ is a parameter representing the position diversity level. Then, partition the kernel into a $\gamma$-almost block diagonal matrix and use BwDPP-MAP to generate the result.

[3]There are rich studies of metrics for CPD problem. The choice of the dissimilarity metric $d(\mathbf{X}_1, \mathbf{X}_2)$ is flexible and could be well tailored according to the characteristics of the data. In our experiments, we use the symmetric KL-divergence and the generalized likelihood ratio (GLR) [10, 11, 12].

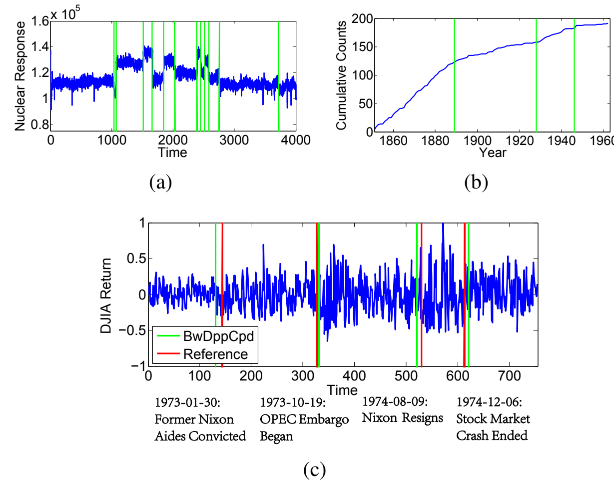



**Fig. 3**. BwDppCpd results for *Well-Log* (a), *Coal Mine Disaster* (b), and *DJIA* (c). Green lines are detected changes.

**Remark 4** *The kernel construction in Step 2 follows the quality-diversity decomposition Eq. (1), where the set of candidates with high quality (high* $d(\mathbf{x}_{t-w+1:t}, \mathbf{x}_{t+1:t+w})$*), and low similarity (distant away from each other), has higher chances to be selected.*

## 3. EXPERIMENTS

In this section, five experiments on real-world time-series data are presented. The first three experiments in Subsection 3.1 examine the algorithm performance on classic CPD testing datasets. We set $\gamma = 0$ because the datasets are small. In the last two experiments, human activity detection and speech segmentation, the DPP kernel sizes are around thousands, making no algorithms capable of performing MAP inference within a reasonable time cost except BwDPP-MAP. We set $\gamma = 3$ for human activity detection and $\gamma = 0, 2$ for speech segmentation. As for the dissimilarity metric $d$, we use Poisson processes and GLR in *Coal Mine Disaster*, and use Gaussian models and SymKL in other experiments [10, 11, 12].

### 3.1. Small-scale Datasets

**Well-Log Data** contains 4050 measurements of nuclear magnetic response taken during the drilling of a well. It is an example of varying Gaussian mean, and the changes reflect the stratification of the earth's crust [9]. Outliers are removed before the experiment. As shown in Fig. 3 (a), all changes are detected by BwDppCpd.

**Coal Mine Disaster Data** [22], a standard dataset for testing CPD method, consists of 191 accidents from 1851 to 1962. The occurring rates of accidents are believed to have changed a few times, and the task is to detect them. The BwDppCpd detection result, as shown in Fig. 3 (b), agrees with that in [7].

**1972-75 Dow Jones Industrial Average Return (DJIA)** contains daily return rates of Dow Jones Industrial Average from 1972 to 1975. It is an example of varying Gaussian variance, where the changes are caused by events that have potential macroeconomic effects. Four changes in the data are detected by BwDppCpd, and are matched well with significant events (Fig. 3 (c)). Compared to [9], one more change is detected (the rightmost), which corresponds to the date that 73-74 stock market crash ended[4]. The result shows

[4]http://en.wikipedia.org/wiki/1973-74_stock_market_crash

| | PRC% | RCL% | $F_1$ |
|---|---|---|---|
| BwDppCpd | 93.05 | 87.88 | 0.9039 |
| RuLSIF | 86.36 | 83.84 | 0.8508 |

**Table 3**. CPD result on human activity detection data *HASC*.

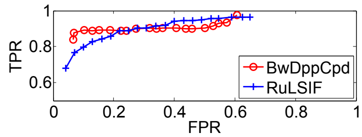



**Fig. 4**. The ROC curve of BwDppCpd and RuLISF.

that the BwDppCpd discovers more information from the data.

### 3.2. Human Activity Detection

*HASC*[5] contains human activity data collected by portable three-axis accelerometers and the task is to locate human behaviour changes. We ran the best algorithm for the dataset, RuLSIF, for comparison. For change-point selection in RuLSIF, the dissimilarity scores are first low-pass filtered so that only points with scores significantly larger than their neighbours may result in peaks. Next, these peaks are identified by thresholding to give the final change points [19].

For evaluation, we first calculated the best precision (PRC), recall (RCL), and $F_1$ score (Table 3), defined as,

$$\mathrm{PRC} = \mathrm{CFC}/\mathrm{DET}, \quad \mathrm{RCL} = \mathrm{CFC}/\mathrm{GT},$$
$$F_1 = 2\,\mathrm{PRC}\,\mathrm{RCL}/(\mathrm{PRC} + \mathrm{RCL}),$$

where CFC is the number of correctly found changes, DET is the number of detected changes, and GT is the number of ground-truth changes. $F_1$ generally reflects PRL and RCL. The result shows BwDppCpd performs generally better.

We also calculated the receiver operating characteristic (ROC) curve (Fig. 4), where true positive rate (TPR) and false positive rate (FPR) are given by $\mathrm{TPR} = \mathrm{RCL}$ and $\mathrm{FPR} = 1-\mathrm{PRC}$. For BwDppCpd, different points are obtained by tuning the position diversity parameter and for RuLSIF by fixing the low-pass filter and tuning parameters for threshold testing. The results show that BwDppCpd outperforms RuLISF when the FPR is low, which should be the area of practical interest in ROC curve.

### 3.3. Speech Segmentation

Speech segmentation is to segment the audio data into acoustically homogeneous segments, e.g. utterances from a single speaker or non-speech portions. We tested two datasets for speech segmentation. The first dataset, *Hub4m97*, is a subset (around 5 hrs) from 1997 Mandarin Broadcast News Speech (HUB4-NE) released by LDC[6]. The second dataset, *TelRecord*, consists of 216 telephone conversations, each around 2-min long, collected from call centres. The two datasets contain utterances with hesitations and a variety of changing background noises, presenting a great challenge for CPD.

We use 12-order MFCCs (Mel-frequency cepstral coefficients) as the time-series data. BwDppCpd with different $\gamma$ for kernel partition (denoted as Bw-$\gamma$ in Table 4) is tested. A classic segmentation method DistBIC [12], a strong baseline for speech segmentation according to our empirical experiments, is used for comparison. In DistBIC, BIC (Bayesian information criterion) dissimilarity scores [11] are first calculated by repeatedly testing single change points along a moving window. Then, the change-point selection is taken by identifying score peaks significantly larger than its neighbouring valleys (Fig. 5), and followed by a BIC-based segment merging procedure. We also use the same merging procedure for BwDppCpd.

[5] http://hasc.jp/hc2011/

[6] http://catalog.ldc.upenn.edu/LDC98S73

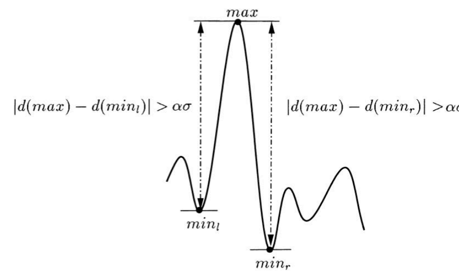



**Fig. 5**. Significant peaks identified by DistBIC. [12]

| | DistBIC | Bw-0 | Bw-2 |
|---|---|---|---|
| *Hub4m97* | | | |
| PRC% | 64.29 | 65.29 | 65.12 |
| RCL% | 74.98 | 78.49 | 78.39 |
| $F_1$ | 0.6922 | 0.7128 | 0.7114 |
| *TelRecord* | | | |
| PRC% | 61.39 | 66.54 | 66.47 |
| RCL% | 81.72 | 85.47 | 84.83 |
| $F_1$ | 0.7011 | 0.7483 | 0.7454 |

**Table 4**. Segmentation results on *Hub4m97* and *TelRecord*.

The experiment results in Table 4 shows that BwDppCpd outperforms DISTBIC in both datasets. Also, comparing the results with $\gamma = 0$ and $\gamma = 2$, using $\gamma = 2$ is faster but gives slightly worse performance. This agrees with our analysis of BwDPP-MAP for using different $\gamma$-partition to trade off speed and accuracy.

## 4. CONCLUSION

In this paper, we introduced BwDPPs, a class of DPPs with almost block diagonal kernels and thus can allow efficient block-wise MAP inference. We use BwDPPs to make change-point selections for CPD problem. The corresponding method, BwDppCpd, showed promising performance in several real-world data experiments.

**Proof of the Argument in Remark 1**: Define

$$\mathbf{S}^i = \begin{cases} \mathbf{L} & i = 0 \\ \begin{bmatrix} \tilde{\mathbf{L}}_{\mathcal{Y}_{i+1}} & [\mathbf{L}_{\mathcal{Y}_{i+1},\mathcal{Y}_{i+2}}\ \mathbf{0}] \\ [\mathbf{L}_{\mathcal{Y}_{i+1},\mathcal{Y}_{i+2}}\ \mathbf{0}]^T & \mathbf{L}_{\cup_{j=i+2}^m \mathcal{Y}_j} \end{bmatrix} & i = 1, \cdots, m-2 \\ \tilde{\mathbf{L}}_{\mathcal{Y}_{i+1}} & i = m-1 \end{cases}.$$

For $i = 1, \cdots, m-1$, $\mathbf{S}^i$ is the Schur complement of $\tilde{\mathbf{L}}_{C_i}$ in $\mathbf{S}^{i-1}_{C_i \cup (\cup_{j=i+1}^m \mathcal{Y}_j)}$, the sub-matrix of $\mathbf{S}^{i-1}$. We next prove the lemma using the first principle of mathematical induction. State the predicate as: $P(i)$: $\mathbf{S}^{i-1}$ and $\tilde{\mathbf{L}}_{\mathcal{Y}_i}$ are positive semi-definite (PSD).

$P(1)$ trivially holds as $\tilde{\mathbf{L}}_{\mathcal{Y}_1} = \mathbf{L}_1$ and $\mathbf{S}^0 = \mathbf{L}$ are PSD.

Assuming $P(i)$ holds. $\mathbf{S}^{i-1}_{C_i \cup (\cup_{j=i+1}^m \mathcal{Y}_j)}$ is PSD because $\mathbf{S}^{i-1}$ is PSD. Assume $\tilde{\mathbf{L}}_{C_i} \succ 0$, which means DPP-MAP does not produce trivial solution. $\mathbf{S}^i$ is the Schur complement of $\tilde{\mathbf{L}}_{C_i}$ in $\mathbf{S}^{i-1}_{C_i \cup (\cup_{j=i+1}^m \mathcal{Y}_j)}$. So $\mathbf{S}^i$ is PSD. Being sub-matrix of $\mathbf{S}^i$, $\tilde{\mathbf{L}}_{\mathcal{Y}_{i+1}}$ is also PSD. Hence, $P(i+1)$ holds.

Therefore, for $i = 1, \cdots, m$, $\tilde{\mathbf{L}}_{\mathcal{Y}_i}$ is PSD.

## 5. REFERENCES


[1] Jinye Zhang, "Speaker segmentation and clustering based on determinantal point processes," *Undergraduate thesis, Tsinghua University*, 2014.

[2] Alex Kulesza and Ben Taskar, "Determinantal point processes for machine learning," *arXiv preprint arXiv:1207.6083*, 2012.

[3] Chun-Wa Ko, Jon Lee, and Maurice Queyranne, "An exact algorithm for maximum entropy sampling," *Operations Research*, vol. 43, no. 4, pp. 684–691, 1995.

[4] Niv Buchbinder, Moran Feldman, Joseph Naor, and Roy Schwartz, "A tight linear time (1/2)-approximation for unconstrained submodular maximization," in *Foundations of Computer Science (FOCS), 2012 IEEE 53rd Annual Symposium on*. IEEE, 2012, pp. 649–658.

[5] Jennifer Gillenwater, Alex Kulesza, and Ben Taskar, "Near-optimal map inference for determinantal point processes," in *Advances in Neural Information Processing Systems*, 2012, pp. 2735–2743.

[6] Byungkon Kang, "Fast determinantal point process sampling with application to clustering," in *Advances in Neural Information Processing Systems*, 2013, pp. 2319–2327.

[7] Peter J Green, "Reversible jump markov chain monte carlo computation and bayesian model determination," *Biometrika*, vol. 82, no. 4, pp. 711–732, 1995.

[8] Marc Lavielle and E Lebarbier, "An application of mcmc methods for the multiple change-points problem," *Signal Processing*, vol. 81, no. 1, pp. 39–53, 2001.

[9] Ryan Prescott Adams and David JC MacKay, "Bayesian online changepoint detection," *arXiv preprint arXiv:0710.3742*, 2007.

[10] Fredrik Gustafsson, "The marginalized likelihood ratio test for detecting abrupt changes," *Automatic Control, IEEE Transactions on*, vol. 41, no. 1, pp. 66–78, 1996.

[11] Scott Chen and Ponani Gopalakrishnan, "Speaker, environment and channel change detection and clustering via the bayesian information criterion," in *Proc. DARPA Broadcast News Transcription and Understanding Workshop*. Virginia, USA, 1998, p. 8.

[12] Perrine Delacourt and Christian J Wellekens, "Distbic: A speaker-based segmentation for audio data indexing," *Speech communication*, vol. 32, no. 1, pp. 111–126, 2000.

[13] Michèle Basseville, Igor V Nikiforov, et al., *Detection of abrupt changes: theory and application*, vol. 104, Prentice Hall Englewood Cliffs, 1993.

[14] Tsuyoshi Idé and Koji Tsuda, "Change-point detection using krylov subspace learning.," in *SDM*. SIAM, 2007, pp. 515–520.

[15] Yoshinobu Kawahara, Takehisa Yairi, and Kazuo Machida, "Change-point detection in time-series data based on subspace identification," in *Data Mining, 2007. ICDM 2007. Seventh IEEE International Conference on*. IEEE, 2007, pp. 559–564.

[16] Frédéric Desobry, Manuel Davy, and Christian Doncarli, "An online kernel change detection algorithm," *Signal Processing, IEEE Transactions on*, vol. 53, no. 8, pp. 2961–2974, 2005.

[17] Takafumi Kanamori, Taiji Suzuki, and Masashi Sugiyama, "Theoretical analysis of density ratio estimation," *IEICE transactions on fundamentals of electronics, communications and computer sciences*, vol. 93, no. 4, pp. 787–798, 2010.

[18] Yoshinobu Kawahara and Masashi Sugiyama, "Sequential change-point detection based on direct density-ratio estimation," *Statistical Analysis and Data Mining*, vol. 5, no. 2, pp. 114–127, 2012.

[19] Song Liu, Makoto Yamada, Nigel Collier, and Masashi Sugiyama, "Change-point detection in time-series data by relative density-ratio estimation," *Neural Networks*, vol. 43, pp. 72–83, 2013.

[20] Luca Guido Molinari, "Determinants of block tridiagonal matrices," *Linear algebra and its applications*, vol. 429, no. 8, pp. 2221–2226, 2008.

[21] Moawwad EA El-Mikkawy, "On the inverse of a general tridiagonal matrix," *Applied Mathematics and Computation*, vol. 150, no. 3, pp. 669–679, 2004.

[22] RG Jarrett, "A note on the intervals between coal-mining disasters," *Biometrika*, vol. 66, no. 1, pp. 191–193, 1979.