
Supplementary material: Rectangular Tiling Process

Masahiro Nakano
Katsuhiko Ishiguro
Akisato Kimura
Takeshi Yamada
Naonori Ueda

NAKANO.MASAHIRO@LAB.NTT.CO.JP
ISHIGURO.KATSUHIKO@LAB.NTT.CO.JP
KIMURA.AKISATO@LAB.NTT.CO.JP
YAMADA.TAK@LAB.NTT.CO.JP
UEDA.NAONORI@LAB.NTT.CO.JP

NTT communication science laboratories, Morinosato Wakamiya 3-1, Atsugi-shi, Kanagawa

1. Examples of projective system

Example 1 (Kingman’s coalescent): Consider a Markov process with values in the space of partitions of $[n]$ ($= \{1, 2, \dots, n\}$), called the n -coalescent. If any two partitions, π, π' , are such that π' can be obtained from π by the coagulation of two of its blocks, then the jump rate from π to π' is 1. All other collections of blocks coagulate at rate 0. That is, given a current state (i.e., the partition) π with k non-empty blocks, the process stays at π for exponential time with parameter $k(k-1)/2$, then jumps at one of the $k(k-1)/2$ partitions, which can be obtained from π by the coagulation of two of its blocks, according to uniform probability. We can verify that the n -coalescent is projective in the following manner. As index set $\mathcal{F}(E)$, we can use the set of all finite partitions of \mathbb{N} . We here use $X_I := 2^{I^2}$, the set of subsets of I^2 . Each $\alpha \in X_I$ is a $(\#I \times \#I)$ binary matrix: $\alpha_{i,j} = 1$ (if $(i, j) \in \alpha$) and $\alpha_{i,j} = 0$ (otherwise), where $\#I$ denotes the number of entries in I . We can define the projection $X_n \rightarrow X_m$ by, for example, the following restriction. For each $1 \leq m \leq n$, let $\varphi_{m,n}$ be the operation on X_n that restricts $\alpha \in X_n$ to the $m \times m$ matrix by keeping the successive $(1:m) \times (1:m)$ entries (in Matlab notation) unchanged, and removing the rest of the entries. For example, consider a draw from $n(=4)$ -coalescent: $1|4|2|3 \rightarrow 14|2|3 \rightarrow 14|23 \rightarrow 1423$. The restricted $m(=3)$ -coalescent corresponds to $1|2|3 \rightarrow 1|23 \rightarrow 123$, that is, “4” is removed by *restriction*. Intuitively, “projective” means that the restricted version can be itself drawn from m -coalescent. For the coagulation $1|4|2|3 \rightarrow 14|2|3$ related to “4”, we can regard that a Poisson process yields an event at rate 6, and it is assigned to candidates related to “4” with probability of $3/6$ (i.e., coagulation candidates $(1, 4), (2, 4), (3, 4)$). This is similar to thinning a Poisson process. That is, its coagulation can be regarded as an event drawn from a Poisson process with rate 3. Coagulations involving “4” are of no concern to the view of $\{1, 2, 3\}$.

Example 2 (Dirichlet process): For example, consider a five-dimensional Dirichlet variable: $(s_1, s_2, s_3, s_4, s_5) \sim$

$\text{Dirichlet}(\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5)$. The key feature of the Dirichlet distribution is *self-similarity*, e.g., $(s_1, s_2, s_3 + s_4, s_5)$ can be regarded as if it were drawn from $\text{Dirichlet}(\alpha_1, \alpha_2, \alpha_3 + \alpha_4, \alpha_5)$. Hence, the Dirichlet process can be constructed by the following projective system. Although the projective limit random probability measures inherently raise some technical difficulties that include measurability and σ -additivity (Orbanz, 2011), here we give priority to intuition. Let G_0 be a base measure on set V . As index set $\mathcal{F}(E)$, we can use the set of all finite partitions of V . To define a partial order on E , let $I = (S_1, \dots, S_m)$ and $J = (S'_1, \dots, S'_n)$, and $I \preceq J : \Leftrightarrow (S_i \cap S'_j)_{i,j} = J$. Intuitively, J shares all the boundaries of I . We can now define the family of $X_I \sim \text{Dirichlet}(G_0(S_1), \dots, G_0(S_m))$ as follows. For each S_i , let $\mathcal{J}_i \subset \{1, \dots, n\}$ be a subset of indices such that $S_i = \cup_{j \in \mathcal{J}_i} S'_j$. We can define the projector as follows: $P_{J,I} X_J(S_i) := \sum_{j \in \mathcal{J}_i} X_J(S'_j)$. This construction is exactly analogous to the above reduction of the five dimensions to four.

2. Formal representation of RTP

Our construction algorithm uses two real values as tunable input parameters, and returns a rectangular partitioning of the input array. Consider a rectangular partitioning of input array I , where I is a sub-array $I \subset \mathbb{N} \times \mathbb{N}$ (i.e., for all integers m, n, m', n' satisfying $1 \leq m \leq m'$ and $1 \leq n \leq n'$, $\{m, m+1, \dots, m'\} \times \{n, n+1, \dots, n'\}$). The rectangular partitioning of I is a collection of disjoint, sub-arrays of I whose union corresponds to I . We write \mathcal{T}_I to denote the collection of all rectangular partitionings of I . The input parameters are a real value, $p \in (0, 1)$, and a budget, $B \geq 0$. Intuitively, $p \in (0, 1)$ directly controls block size, that is, if we choose a large value for p , each block is expected to be large.

Our algorithm consists of two stages: “generating order of block growth” and “assigning entries to blocks”. These cor-

Algorithm 1 SAMPLING ORDER OF BLOCK GROWTH

SAMPLEM(X, t, θ_I)
 · $\xi \sim \text{Exponential}(c(\{X\}))$, where $c(\cdot)$ denotes a rate function (that is typically a counter of the sum of the number of columns and rows).
if $t < \xi$ **then**
 · Draw a uniform binary indicator: $l_X \sim \text{Bernoulli}(1/2)$.
 if $l_X = 0$ **then**
 · Update θ_I by adding new relations $(u, v) \prec (u', v')$ for every pairs $\{(u, v), (u', v') \mid (u, v), (u', v') \in X, v < v'\}$.
 else
 · Update θ_I by adding new relations $(u, v) \prec (u', v')$ for every pairs $\{(u, v), (u', v') \mid (u, v), (u', v') \in X, u < u'\}$.
 end if
 · **return** trivial partition $\{X\}$ and θ_I .
else
 · Sample a uniform axis-parallel partition $\{X_0, X_1\}$ of X .
 if There are $x_0 \in X_0, x_1 \in X_1$ such that $x_0 \prec x_1$ **then**
 · Update θ_I by adding relations $x_0 \prec x_1$ for every pairs $x_0 \in X_0, x_1 \in X_1$.
 else
 · Update θ_I by adding relations $x_0 \succ x_1$ for every pairs $x_0 \in X_0, x_1 \in X_1$.
 end if
 · **return** $\cup_{i \in 2} \text{SAMPLEM}(X_i, t - \xi, \theta_I)$.
end if

respond to the Bayesian hierarchy, that is, the former means conditioning, and the latter generates a sample based on a conditional probabilistic model. Plainly, for the first step, our algorithm generates the total order of all entries of I , denoted by $\theta_I := (I; \prec)$, by applying a discrete Mondrian process with budget B . More precisely, we denote by Θ_I the set, in ascending order of $(m' - m + 1)(n' - n + 1)$, of entries of I , i.e., for any $1 \leq i < j \leq (m' - m + 1)(n' - n + 1)$, $\theta_I(i) \prec \theta_I(j)$; the first step generates sample θ_I of Θ_I . This step is recursively processed. As the initialization, partial clues of θ_I are given: for any natural numbers u and $v < v'$, $(u, v) \prec (u, v')$, and for any natural numbers v and $u < u'$, $(u, v) \prec (u', v)$. We recursively run $\text{SAMPLEM}(I, T, \theta_I)$ based on Algorithm 1. It returns a sample of the total order of all entries of I , i.e., θ_I . For the second step, our algorithm assigns the entries to blocks in the obtained order, θ_I , using two types of coins whose probabilities of turning up heads when tossed are p and q such that $q = p/(p^2 - p + 1)$. We run $\text{SAMPLER}(I, \theta_I, p)$ based on Algorithm 2. Finally, we obtain a rectangular partitioning of I .

Algorithm 2 SAMPLING RECTANGULAR PARTITIONING

SAMPLER(I, θ_I, p)
 · Start from the $\theta_I(1)$ -entry as a singleton.
for $i = 2$ to $(m' - m + 1)(n' - n + 1)$ **do**
 if Two adjacent entries $\theta_I(i) + (-1, 0)$ and $\theta_I(i) + (0, -1)$ exist in I , and they have already been assigned to blocks **then**
 if The two adjacent entries are assigned to the same block **then**
 · The $\theta_I(i)$ -entry is also assigned to the same block.
 else
 if $R_I(i) + (-1, 0) \prec \theta_I(i) + (1, 0)$ **then**
 · With probability p , the $\theta_I(i)$ -entry is assigned to the block to which the $\theta_I(i) + (-1, 0)$ -entry belongs. With probability $(1 - p)q$, it is assigned to the block to which the $\theta_I(i) + (0, -1)$ -entry belongs. With probability $(1 - p)(1 - q)$, it is assigned to a new block.
 else
 · With probability p , the $\theta_I(i)$ -entry is assigned to the block to which the $\theta_I(i) + (0, -1)$ -entry belongs. With probability $(1 - p)q$, it is assigned to the block to which the $\theta_I(i) + (-1, 0)$ -entry belongs. With probability $(1 - p)(1 - q)$, it is assigned to a new block.
 end if
 end if
 end if
 · With probability p , the $\theta_I(i)$ -entry is assigned to the same block to the adjacent entry; with probability $(1 - p)$, to a new block.
 end if
end for
return the resulting rectangular partitioning.

Our algorithm provides a self-consistent family of rectangular partitionings of matrices of any finite size, which leads to a probability measure on rectangular partitionings of matrices of infinite size.

3. Proof of theorems and propositions

Proof of proposition 3.2

Without loss of generality, we assume that the input of RLGA is $\{\rightarrow, \downarrow\}$. First, we focus on LGA, i.e., the rectangular partitioning of (2×2) -arrays. Note that the top row (two entries) was first to be assigned to blocks, and then the bottom-left entry was assigned. More precisely, we first decide whether or not the top-left block gains an increment in the horizontal direction with $\text{Bernoulli}(p)$, then whether or not it gains one in the vertical direction with $\text{Bernoulli}(p)$.

By construction, we can easily check that (1-i) the probability of the top-right and bottom-right belonging to the same block is p , and (1-ii) the probability of the bottom-left and bottom-right belonging to the same block is also p . For (1-i), by marginalizing out all possibilities of the top-left and the bottom right entries, we can calculate the probability that the top-right and bottom-right belong to the same block:

$$\begin{aligned} & p^2 + (1-p)pq + (1-p)^3q \\ &= p^2 + \frac{p}{p^2 - p + 1}(1-p)(p^2 - p + 1) = p. \end{aligned} \quad (1)$$

For (1-ii), by marginalizing out all possibilities of the top-left and top-right entries, we can calculate the probability that the bottom-left and bottom-right belong to the same block:

$$p(1-p)p + p^2 + (1-p)^2p = p. \quad (2)$$

Finally we have to show that (i) the vertical length can be incremented by 1 with probability p , and (ii) the horizontal length can also be incremented by 1 with probability p . For (i), by construction and the above result, the vertical length is incremented by 1 with probability p . For (ii), the horizontal length should be decided when its first column is grown. Thus, by construction and the above result, the horizontal length also increments by 1 with probability p .

Proof of Theorem 3.3

Without loss of generality, we assume that the input of RLGA is $\rightarrow\downarrow$. We have to consider four types of restriction, (1) deleting the right column, (2) deleting the bottom row, (3) deleting the left column, and (4) deleting the top row. (1) and (2) can be easily checked by construction, that is, the deleted row or column are generated based on the other part of the rectangular partitioning. For (3), we consider repeated marginalization from the top-left (2×2) -array to the bottom-left (2×2) -array. When we marginalize the left column of each (2×2) -array, we can consider incrementing the right column by 1 with probability p . Thus, we can recursively check that the process is self-consistent under the condition that the left column is deleted.

For (4), we have to check whether we can regard that the restricted (remained) top row is incremented with probability p . Each block of the restricted top row are originally (i.e., before restriction) generated in the following two manners: (i) original top column generated the block, and was then incremented, or (ii) it is originally generated in the restricted top column for the first time. Thus, when we marginalize all possible patterns of the original top column, we can regard each block of the restricted top column as being capable of being incremented vertically with probability p . That is, it shows self-consistency when the top column is deleted.

Algorithm 3 MCMC FOR RTP-BASED RELATIONAL MODEL

Input: Observation $(Y_{i,j})_{m \times n}$, p ($0 < p < 1$) and *global direction* (as an example, \searrow)

Initialization

- Generate a regular grid partitioning (i.e., a pre-cluster) of $[0, 1]^2$ based on the vertical and horizontal PPs on $[0, 1]$ (in our experiment, we manually give this step).
- For each row i , generate the coordinate from Uniform $[0, 1]$, and initialize ζ_i .
- For each column j , generate the coordinate from Uniform $[0, 1]$, and initialize η_j .
- Generate a RTP partitioning of the pre-cluster from the RTP. This step involves MP partitioning and directions of growth.

Main loop of MCMC

- Update the PPs based on random-walk MH.
 - For each row i , update ζ_i using Gibbs sampling, similar to (Roy & Teh, 2009) and (Wang et al., 2011).
 - For each column j , update η_j using Gibbs sampling, similar to (Roy & Teh, 2009) and (Wang et al., 2011).
 - Update the MP partitioning using a reversible jump method, similar to (Wang et al., 2011).
 - For each box in MP partitioning, update the direction of growth using Gibbs sampling.
 - Update the RTP rectangular partitioning based on MH FOR RTP (Algorithm 4).
-

Proof of theorem 3.4

Recall that Θ_I consists of a hierarchical partitioning of I drawn from the discrete Mondrian process, and directions of growth. This discrete Mondrian process provides a projective system of hierarchical partitionings, which can be verified as being similar to the self-similarity of the original Mondrian process ((Roy, 2011), Prop. V. 10). Moreover, each direction of growth is independently chosen. Thus, Θ_I ($I \in \mathcal{F}(E)$) are projective. As a result, it follows from Theorem 3.2 and the property of RLGA that the family of $\mu^I(T_I|\Theta_I)$ for $I \in \mathcal{F}(E)$ is conditionally projective, and leads to the conditional projective limit $\mu^E(T_E|\Theta_E)$.

4. Inference

We use Markov chain Monte Carlo (MCMC) methods that iterate over draws from posteriors to yield the rectangular partitioning T , the pre-clusters ξ and η , and the intermediate variables θ . Note that we can easily marginalize out the Dirichlet variables ϕ , in the same way as (Wang et al., 2011). Algorithm 3 provides a sketch.

Rectangular partitioning T : It is not easy to sample the conditional posterior distribution for rectangular partitioning in the sense of Gibbs sampling. Thus, we employ a

Algorithm 4 MH FOR RTP

-
- Choose uniformly at random one row/column from the matrix, i.e., the pre-cluster. For example, in the following, suppose that a row is chosen.
 - Generate uniformly at random an order of priorities of “left”, “upper”, and “lower”. For example, in the following, we assume that “left” \succ “upper” \succ “lower”.
 - for** from left-most entry to right-most entry of the chosen row **do**
 - Generate a real value $\gamma \in [0, 1]$ from Uniform $[0, 1]$. If possible (i.e., satisfies the rectangular partitioning constraint), the current entry is assigned to the “left” block. Otherwise, go to the next line.
 - Regenerate real value $\gamma \in [0, 1]$ from Uniform $[0, 1]$. If possible, the current entry is assigned to the “upper” block. Otherwise, go to the next line.
 - Regenerate real value $\gamma \in [0, 1]$ from Uniform $[0, 1]$. If possible, the current entry is assigned to the “lower” block. Otherwise, go to the next line.
 - The current entry is assigned to a new block.
 - end for**
 - Applying the acceptance/rejection scheme of the MH.
-

Metropolis-Hasting (MH) algorithm: we generate the next candidate from a proposal distribution, and then accept or reject it based on the probability ratio. In terms of the MH scheme, it is important to design a good (ideally, rapid mixing and high acceptance rate) proposal. Our strategy is to change the partitions of only one row/column per iteration, and keeping the remainder. In our experiments, we used the following proposal: (as an example, for one row) from left to right, holding the constraint of the rectangular tilings, we randomly chose whether an entry is assigned to the adjacent upper, lower, left, or new block (Algorithm 4).

Pre-clusters ξ and η : Precisely, this step involves not only pre-clusters but also the split locations of the (vertical and horizontal) Poisson processes. For the pre-clusters, we can use Gibbs sampling. This is due to an artifice of the combination of PPs and RTP. PPCs make it possible to separate the updates of the permutations (i.e., assigning the pre-clusters) from that of the rectangular partitioning. Otherwise, the updates of the permutations influence the rectangular partitioning. For the split locations of PPs, we can use random walk MH.

Intermediate variables θ : We here discuss a sampler for θ consisting of the hierarchical partitioning \mathcal{M} and the binary variables of leaf blocks. For the binary variables, we can easily use Gibbs sampling. For MP sample \mathcal{M} (consisting of a tree of sub-arrays, random costs and locations of the cut to each non-leaf block of the tree), we can employ a reversible jump MCMC (Wang et al., 2011).

5. Properties of RLGAs

We here analyze RLGAs more carefully. Specifically, we clarify the influence of the block growth direction. In this appendix, we extend index set E to the product of integers \mathbb{Z}^2 . Note that this extension does not affect our main construction. There are eight possible block growth patterns of RLGA, i.e., a combination of four corners as starting points and two choices of l . For understanding we introduce our notations. We use two random binary variables o_{hor} and o_{ver} to express four choices of starting points for the four patterns of $(o_{\text{hor}}, o_{\text{ver}})$. They provide partial clues of the total order $(I; \prec)$: $o_{\text{hor}} = 0/1$ means that, for any integers u and $v < v'$, $(u, v) \prec / \succ (u, v')$. Similarly, $o_{\text{ver}} = 0/1$ means that, for any integers v and $u < u'$, $(u, v) \prec / \succ (u', v)$. More intuitively, we also write eight patterns as follows:

- $\rightarrow\downarrow$: $o_{\text{hor}} = 0, o_{\text{ver}} = 0, l = 0$.
- $\downarrow\rightarrow$: $o_{\text{hor}} = 0, o_{\text{ver}} = 0, l = 1$.
- $\leftarrow\downarrow$: $o_{\text{hor}} = 1, o_{\text{ver}} = 0, l = 0$.
- $\downarrow\leftarrow$: $o_{\text{hor}} = 1, o_{\text{ver}} = 0, l = 1$.
- $\rightarrow\uparrow$: $o_{\text{hor}} = 0, o_{\text{ver}} = 1, l = 0$.
- $\uparrow\rightarrow$: $o_{\text{hor}} = 0, o_{\text{ver}} = 1, l = 1$.
- $\leftarrow\uparrow$: $o_{\text{hor}} = 1, o_{\text{ver}} = 1, l = 0$.
- $\uparrow\leftarrow$: $o_{\text{hor}} = 1, o_{\text{ver}} = 1, l = 1$.

A natural question is whether the probabilities that a given rectangular partitioning of input array $I, T_I \in \mathcal{T}_I$, is drawn from eight patterns are equal or not. We have the following statement:

Theorem A. 1

Let T_I be a rectangular partitioning of I drawn from an RLGA with $o_{\text{hor}}, o_{\text{ver}}, l_I$: $T_I \sim \text{RLGA}(o_{\text{hor}}, o_{\text{ver}}, l_I)$ (e.g., $\rightarrow\downarrow$). (1) It is distributionally equivalent to $T_I \sim \text{RLGA}((1-l_I)o_{\text{hor}} + l_I(1-o_{\text{hor}}), (1-l_I)(1-o_{\text{ver}}) + l_I o_{\text{ver}}, l_I)$ (e.g., $\rightarrow\uparrow$). (2) It cannot be regarded as if $T_I \sim \text{RLGA}(l_I o_{\text{hor}} + (1-l_I)(1-o_{\text{hor}}), l_I(1-o_{\text{ver}}) + (1-l_I)o_{\text{ver}}, l_I)$ (e.g., $\leftarrow\downarrow$).

Proof Without loss of generality, we assume that the input of RLGA is $o_{\text{hor}} = 0, o_{\text{ver}} = 0, l_I = 0$. We also assume, w.l.o.g., $I = 1 : m \times 1 : n$ in Matlab notation. For notational simplicity, we express the probability that T_I is drawn from RTP($\rightarrow\downarrow$) as $\Pr_{\rightarrow\downarrow}[T_I]$. Moreover, we write a rectangular partitioning of the restriction of I as, for example, $T_{I(1:m', \cdot)}$ in Matlab notation.

We prove the first statement by induction. By construction, we have

$$\Pr_{\rightarrow\downarrow}[T_I] = \Pr_{\rightarrow\downarrow}[T_{I(2,:)}|T_{I(1,:)}] \cdot \Pr_{\rightarrow\downarrow}[T_{I(3,:)}|T_{I(2,:)}] \\ \times \cdots \times \Pr_{\rightarrow\downarrow}[T_{I(m,:)}|T_{I(m-1,:)}]. \quad (3)$$

Thus, we have only to deal with the case of two columns, which can be easily extended to any finite-dimensional array. The base case, (2×2) -arrays, can be easily checked. Suppose that, for $(2 \times n')$ -arrays, $\text{RLGA}(0, 0, 0)$ is distributionally equivalent to $\text{RLGA}(0, 1, 0)$. For $(2 \times n' + 1)$ -arrays, we can easily check that all possible patterns have the same probabilities from LGA.

For the second statement, we can easily find counterexamples, for example, (2×2) -arrays.

Although the second statement of the above proposition is unfortunate, we may intuitively think that, for example, $\text{RLGA}(\rightarrow\downarrow)$ has a certain resemblance to $\text{RLGA}(\leftarrow\downarrow)$. This intuition is partially correct. In the following, we discuss it more carefully. We begin with the following remark, and then move to a stronger statement.

Remark A. 2 Let I, J, K be the sub-arrays of \mathbb{N}^2 such that $I = \{m + 2, \dots, m'\} \times \{n + 2, \dots, n'\}$, $J = \{m + 1, \dots, m' + 1\} \times \{n + 1, \dots, n' + 1\}$, and $K = \{m, \dots, m' + 2\} \times \{n, \dots, n' + 2\}$. Let $B_K|_{T_J}$ be a set of rectangular partitionings of K where its restriction to J is $T_J \in \mathcal{T}_J$ and the entries in J do not belong to any blocks that the entries in $K \setminus J$ belong to. The sum of the probability that the partitionings in $B_K|_{T_J}$ are drawn from $\text{RLGA}(o_{\text{hor}}, o_{\text{ver}}, l_I)$ (e.g., $\rightarrow\downarrow$) is equal to the sum of the probabilities that they are drawn from $\text{RLGA}(l_I o_{\text{hor}} + (1 - l_I)(1 - o_{\text{hor}}), l_I(1 - o_{\text{ver}}) + (1 - l_I)o_{\text{ver}}, l_I)$ (e.g., $\leftarrow\downarrow$). For any $T_I \in \mathcal{T}_I$, there is a set $B_K|_{T_J}$ whose restriction to I corresponds to T_I .

Although the above remark actually provides some insights, it is nothing more than a special property of the compartments of rectangular partitionings. Plainly, that remark covers only the partitionings that have large rectangles whose edges consists only of the edges of blocks. Thus, it holds only in very special cases. We have to consider a common property of all rectangular partitionings. For simplicity, with loss of generality, we focus on the relation between the case $o_{\text{hor}} = 0, o_{\text{ver}} = 0, l_I = 0$ and the case $o_{\text{hor}} = 1, o_{\text{ver}} = 0, l_I = 0$. Our strategy is to approximate any finite sub-array I of any rectangular partitioning $T_E \in \mathcal{T}_E$ by a compartment of T_E that covers I and can be regarded as a drawn from both $\text{RLGA}(0, 0, 0)$ and $\text{RLGA}(1, 0, 0)$. To state this more precisely, we introduce a specific definition of compartment:

Definition A. 3

For any sub-arrays I of \mathbb{Z}^2 , let \mathcal{U}_{T_I} be a set of rectangular partitionings where

- their restrictions to I correspond to $T_I \in \mathcal{T}_I$,
- for any row, both the leftmost entries and the rightmost entries are on edges of blocks, and
- for any two adjacent rows, the horizontal difference value of their leftmost entries is equal to the horizontal difference value of their rightmost entries.

We call set \mathcal{U}_{T_I} the investiture of T_I . Moreover, we write $\mathcal{U}_{T_I}^*$ as the set satisfying only the first and second arguments.

We have the following theorem:

Theorem A. 4

For any sub-arrays $I = (m : m') \times (n : n')$ of \mathbb{Z}^2 , the probability that $U_{T_I} \in \mathcal{U}_{T_I}$ is drawn from $\text{RLGA}(0, 0, 0)$ is equal to the probability that U_{T_I} is drawn from $\text{RLGA}(1, 0, 0)$. Moreover, for any finite-dimensional sub-arrays I of \mathbb{Z}^2 and infinite-dimensional rectangular partitionings $T_E \in \mathcal{T}_E$, there exists, with probability one, $U_{T_E|I} \in \mathcal{U}_{T_E|I}$.

To prove the above proposition, we introduce the following three lemmas.

Lemma A. 5

For sub-array $I = (m : m') \times (n : n')$ and integer sequences $\mathbf{g} = (g_1, g_2, \dots, g_{m'-m+1})$, $\mathbf{h} = (h_1, h_2, \dots, h_{m'-m+1})$, the probability that $T_{\cup_{u=m}^{m'} I(u, g_u : h_u)}$ is drawn from $\text{RLGA}(\rightarrow\downarrow)$ can be expressed as follows:

$$\Pr_{\rightarrow\downarrow}[T_{\cup_{u=m}^{m'} I(u, g_u : h_u)}] \\ = \prod_{u=m}^{m'-1} \Pr_{\rightarrow\downarrow}[T_{I(u+1, g_{u+1} : h_{u+1})} | T_{I(u, g_u : h_u)}]. \quad (4)$$

Proof By construction, we can easily check that the $m'+1$ -th row only depends on the m' -th rows.

To state the following lemma, we introduce $e(u, v : v + 1)$ which means that the (u, v) -entry belongs to blocks different from those that the $(u, v+1)$ -entry belongs to. Thus, we write $\Pr[e(u, v : v + 1)]$ as the probability that the (u, v) -entry and the $(u, v + 1)$ -entry belong to different blocks.

Lemma A. 6

For sub-array $I = (m : m') \times (n : n')$ and integer sequences $\mathbf{g} = (g_m, g_{m+1}, \dots, g_{m'})$, such that, any $u = m, \dots, m'$, $n < g_u < n'$,

$$\Pr_{\rightarrow\downarrow}[\cap_u e(u, g(u) : g(u) + 1)] = \\ \Pr_{\leftarrow\downarrow}[\cap_u e(u, g(u) : g(u) + 1)]. \quad (5)$$

Proof It follows from Theorem 4.1 that, for any $u = m, m + 1, \dots, m'$, probability $\Pr_{\rightarrow\downarrow}[e(u, g(u) : g(u) + 1)]$

is equivalent to probability $\Pr_{\downarrow\leftarrow}[e(u, g(u) : g(u) + 1)]$. Moreover, by construction, $\Pr_{\rightarrow\downarrow}[e(u + 1, g(u + 1) : g(u + 1) + 1) \mid e(u, g(u) : g(u) + 1)]$. As a result, we have

$$\begin{aligned}
 & \Pr_{\rightarrow\downarrow}[\cap_u e(u, g(u) : g(u) + 1)] \\
 &= \prod_{u=m}^{m'} \Pr_{\rightarrow\downarrow}[e(u + 1, g(u + 1) : g(u + 1) + 1) \\
 &\quad \mid e(u, g(u) : g(u) + 1)] \\
 &= \prod_{u=m'}^m \Pr_{\rightarrow\uparrow}[e(u, g(u) : g(u) + 1) \\
 &\quad \mid e(u + 1, g(u + 1) : g(u + 1) + 1)] \\
 &= \prod_{u=m}^{m'} \Pr_{\leftarrow\uparrow}[e(u + 1, g(u + 1) : g(u + 1) + 1) \\
 &\quad \mid e(u, g(u) : g(u) + 1)] \\
 &= \prod_{u=m'}^m \Pr_{\leftarrow\downarrow}[e(u, g(u) : g(u) + 1) \\
 &\quad \mid e(u, g(u + 1) : g(u + 1) + 1)] \\
 &= \Pr_{\leftarrow\downarrow}[\cap_u e(u, g(u) : g(u) + 1)] \quad . \quad (6)
 \end{aligned}$$

This completes the proof.

We need the following third lemma to prove Theorem A.4.

Lemma A.7

For a pair of sub-arrays $I = (m : m') \times (v : v')$ and $J = (m : m') \times (n : n')$ such that $n < v < v' < n'$, if all widths of blocks are less than $(v - n + n' - v'/4)^{1/(m+1)}$, then there are $U_{T_I} \in \mathcal{U}_{T_I}$ that are covered by J .

Proof We employ the pigeonhole principle. We divide $J \setminus I$ into subsets each with s -length width. If any block does not grow horizontally more than s , then, for any rows, least one block edge is present. Thus, the number of possible patterns of lines joining the entries on edges is at most s^{m+1} . We here have

$$\begin{aligned}
 & s \left\lceil \frac{v - n}{s} \right\rceil + s \left\lceil \frac{n' - v'}{s} \right\rceil > s(s^m + 1) \\
 \Leftarrow v - n - s + n' - v' - s & > s^{m+1} + s \\
 \Leftarrow v - n + n' - v' & > s^{m+1} + 3s \\
 \Leftarrow v - n + n' - v' & > 4s^{m+1} \\
 \Leftarrow s & < \left(\frac{v - n + n' - v'}{4} \right)^{\frac{1}{m+1}} \quad (7)
 \end{aligned}$$

where $\lceil x \rceil$ denotes the integer part of real value x . As a result, if we have more than $s^m + 1$ subsets, then there exists at least one pair of the same patterns based on the pigeonhole principle. This completes the proof.

We now prove Theorem A.4.

Proof To prove the former statement, we first show that, for sub-array $I = (m : m') \times (n : n')$ and integer sequences $\mathbf{g} = (g_m, g_{m+1}, \dots, g_{m'})$ and $\mathbf{h} = (h_m, h_{m+1}, \dots, h_{m'})$, we have

$$\begin{aligned}
 & \Pr_{\rightarrow\downarrow}[T_{\cup_{m'} I(m', 1 : g_{m'})}] \\
 &= \prod_{m'=1}^{m-1} \Pr_{\rightarrow\downarrow}[T_{I(m'+1, 1 : g_{m'+1})} \mid T_{I(m', 1 : g_{m'})}] \\
 &= \prod_{m'=1}^{m-1} \Pr_{\leftarrow\uparrow}[T_{I(m', n - g_{m'} : n)} \mid T_{I(m'+1, n - g_{m'+1} : n)}] \\
 &= \prod_{m'=1}^{m-1} \Pr_{\leftarrow\downarrow}[T_{I(m', n - g_{m'} : n)} \mid T_{I(m'+1, n - g_{m'+1} : n)}] \\
 &= \Pr_{\leftarrow\downarrow}[T_{\cup_{m'} I(m', 1 : g_{m'})}] \quad . \quad (8)
 \end{aligned}$$

Hereafter we can easily complete the proof of the former statement by induction on the increment of the width that does not belong to blocks that have at least one entry that is on the edge. This completes the proof of the first statement.

For the latter statement, we evaluate the probability that every union of blocks in T_E does not belong to \mathcal{U}_{T_I} . For any sub-arrays $J = (m : m') \times (n : n')$ of \mathbb{Z}^2 such that $n < v < v' < n'$, we consider random variable $\chi_J : \mathcal{T}_E \rightarrow \{0, 1\}$ where, if $T_E|_J$ has $U_{T_I} \in \mathcal{U}_{T_I}$, then $\chi_J = 1$; otherwise, $\chi_J = 0$. It follows from Lemma 4. 7 that at least one block has width more than $(v - n + n' - v'/4)^{1/(m+1)}$ -length. Thus we have

$$\begin{aligned}
 \Pr_{\rightarrow\downarrow}[\chi_J = 0] &= \sum_{U_{T_I} \in \mathcal{U}_{T_I}^*} \Pr_{\rightarrow\downarrow}[U_{T_I} \notin \mathcal{U}_{T_I}] \\
 &< p^{(v - n + n' - v'/4)^{1/(m+1)}} \quad (9)
 \end{aligned}$$

As a result, if J is sufficiently large, i.e., $n' - n \rightarrow \infty$, then $\chi_J = 1$ with probability one. Similarly, we check the case of $\leftarrow\downarrow$. This completes the proof of the second statement.

6. Experimental result

We here show some figures omitted from the main body.

Visualization of partitioning. (a) **Animal feature.** Fig. 1 shows samples of RTP-based analysis with animal-feature labels. (b) **Dnations.** This data was hard for the MP. Typically, the MP runs fall into a trivial partition (i.e., no cut). Thus, Fig. 2 shows samples of the RTP- and the IRM-based analysis. (c) **Cities.** Fig. 3 shows samples of the RTP- and the MP-based analysis.

Rectangular Tiling Process

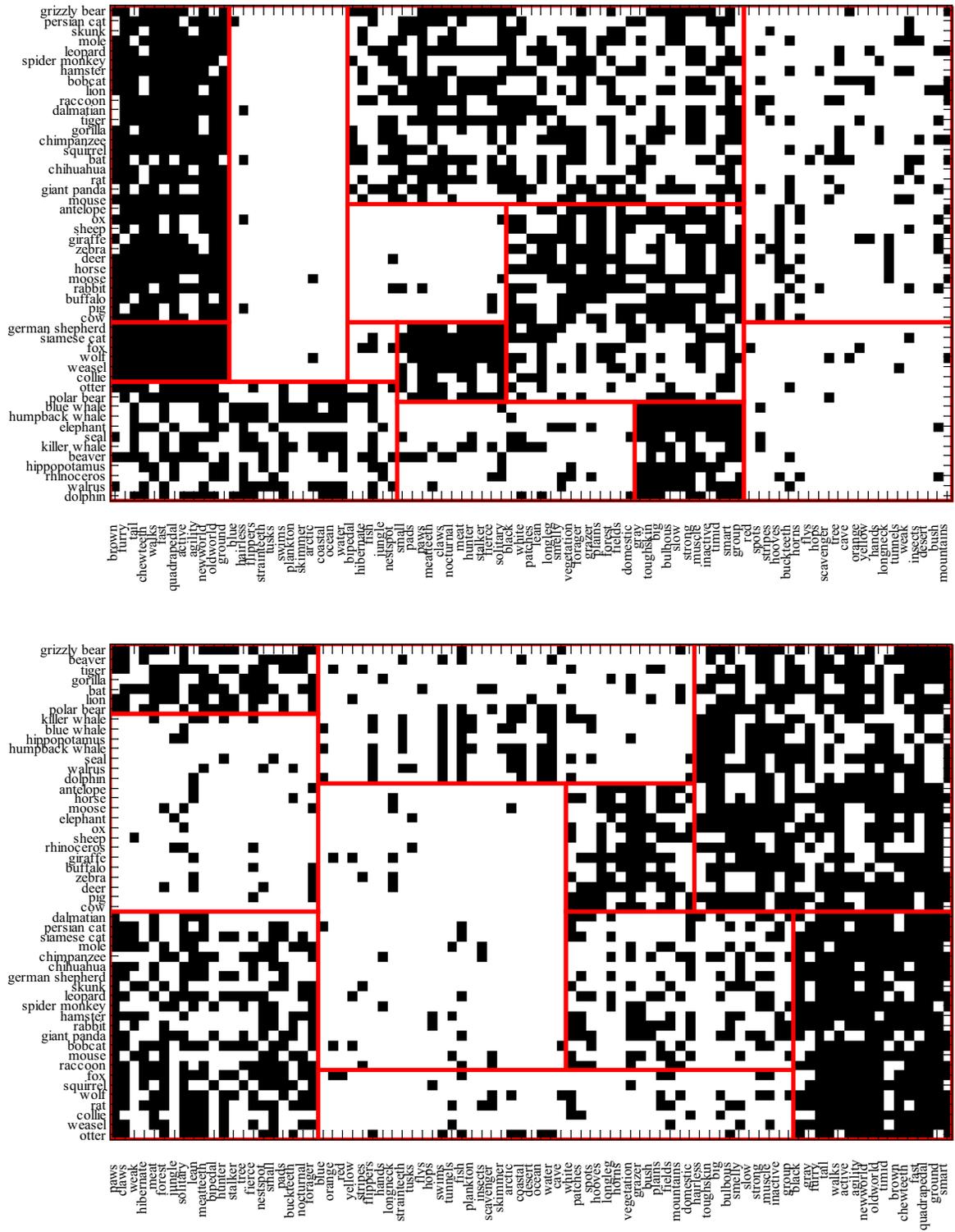


Figure 1. Animal-feature data analysis. Two samples of RTP-based analysis. As an example, for the right analysis of the RTP, two dense blocks on the right side indicates that {active, fast, smart} are included in a cluster for all animals, but {big, strong, group} are included in a cluster only for approximately half of animals.

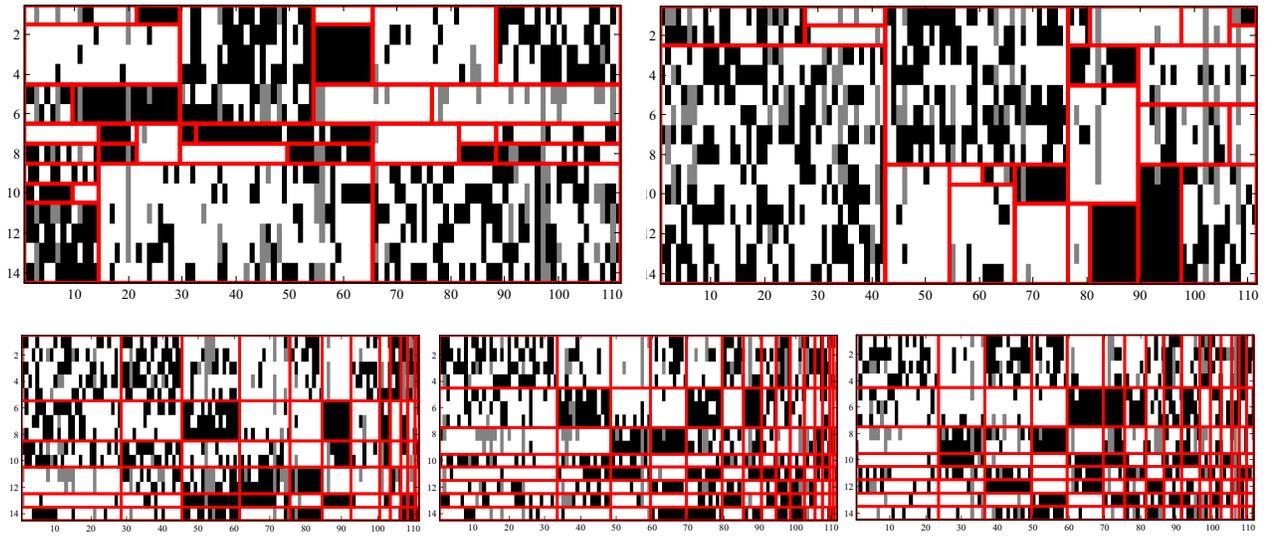


Figure 2. Dnations data analysis. **Top:** two samples of RTP-based analysis. **Bottom:** three samples of IRM-based analysis.

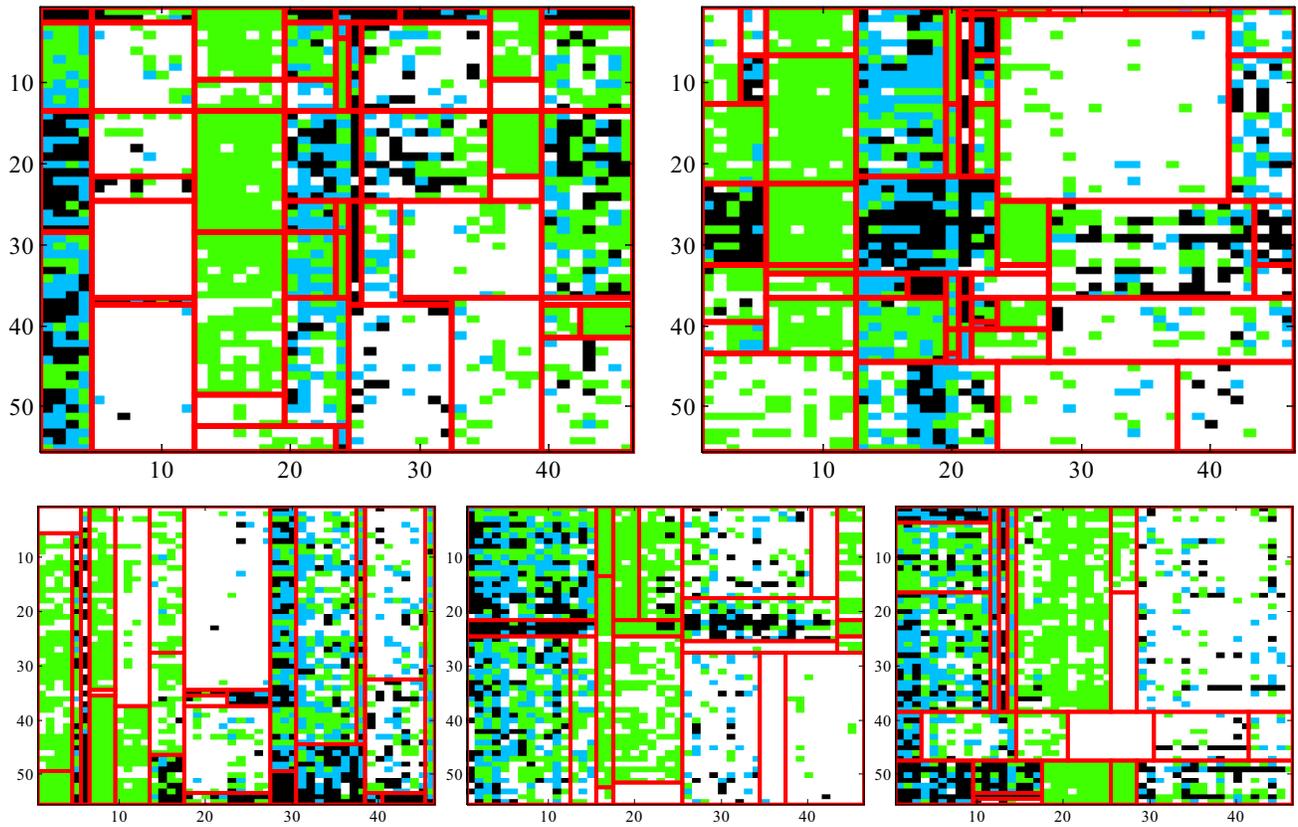


Figure 3. Cities data analysis. **Top:** two samples of RTP-based analysis. **Bottom:** three samples of MP-based analysis.

References

- Orbanz, P. Projective limit random probabilities on polish spaces. *Electronic Journal of Statistics*, 5, 2011.
- Roy, D. M. *Computability, inference and modeling in probabilistic programming*. PhD thesis, Massachusetts Institute of Technology, 2011.
- Roy, D. M. and Teh, Y. W. The Mondrian process. In *Advances in Neural Information Processing Systems*, 2009.
- Wang, P., Laskey, K. B., Domeniconi, C., and Jordan, M. I. Nonparametric bayesian co-clustering ensembles. In *SIAM International conference on Data Mining*, 2011.