Comput Stat (2013) 28:831–850
DOI 10.1007/s00180-012-0331-3

ORIGINAL PAPER

MCMC using Markov bases for computing p-values in decomposable log-linear models

Masahiro Kuroda · Hiroki Hashiguchi · Shigekazu Nakagawa · Zhi Geng

Received: 26 January 2011 / Accepted: 18 April 2012 / Published online: 3 June 2012

Abstract We derive an explicit form of a Markov basis on the junction tree for a decomposable log-linear model. Then we give a description of a Markov basis characterized by global Markov properties associated with the graph of a decomposable log-linear model and show how to use the Markov basis for generating contingency tables of a Markov chain. The estimates of exact p-values can be obtained from contingency tables generated from the proposed Markov chain Monte Carlo using the Markov basis.

Keywords Decomposable log-linear models · Junction tree · Markov basis · Markov chain Monte Carlo · p-value

1 Introduction

Log-linear models are the general way of studying contingency tables. To measure how well a log-linear model fits to the data in a contingency table, we use asymptotic tests based on large sample approximations such as chi-squared approximations.

M. Kuroda (✉)
Department of Socio-Information, Okayama University of Science,
1-1 Ridaicho, Kita-ku, Okayama 700-0005, Japan
e-mail: kuroda@soci.ous.ac.jp

H. Hashiguchi
Department of Information and Computer Sciences, Saitama University,
255 Shimo-Okubo, Sakura, Saitama 338-8570, Japan

S. Nakagawa
Department of Computer Science and Mathematics, Kurashiki University of Science and the Arts,
2640 Nishinoura, Tsurajima-cho, Kurashiki-shi, Okayama 712-8505, Japan

Z. Geng
Institute of Mathematical Sciences, Peking University, Beijing 100871, China

For large and sparse contingency tables, exact tests evaluating statistical significance using p-values are preferable to asymptotic tests; however, the computation of an exact p-value for testing a log-linear model may be infeasible due to the enumeration of all contingency tables with fixed marginals specified by the log-linear model. To study such tables, Markov chain Monte Carlo (MCMC) is a popular technique for generating contingency tables with fixed marginals. Diaconis and Sturmfels (1998) suggested the computation of a Markov basis by finding Gröbner bases, thus allowing the construction of a Markov chain that connects all contingency tables with fixed marginals.

Graphical models are statistical models for interpreting conditional independence relationships among a number of variables and are a subclass of the general class of hierarchical log-linear models, as described in Lauritzen (1996). Decomposable graphical models, a subclass of graphical models, are important in applied graphical modeling, because numerous theoretical and computational aspects are more useful and tractable with decomposable graphical models than with general graphical models. Moreover, an attractive feature of decomposable models is their ease of interpretation. Dobra (2003) gave explicit formulae of Markov bases for decomposable log-linear models and described the algorithm for generating Markov bases. Geiger et al. (2006) analyzed the algebraic properties of decomposable graphical models and provided various characterizations of the models. Hara et al. (2010) clarified structures of Markov bases for decomposable models and gave a complete description of minimal and minimal invariant Markov bases for the models.

In this paper, we give a new explicit formula that identifies a Markov basis on the junction tree for a decomposable log-linear model and show MCMC using the Markov basis for computing the p-value for a decomposable log-linear model. In Sect. 2, we define decomposable log-linear models and the junction trees constructed by the graphs of decomposable log-linear models. In Sect. 3, we give a description of a Markov basis characterized by the global Markov properties of Lauritzen (1996) on the graph of a decomposable log-linear model. In Sect. 4, we show how to employ the Markov basis for generating contingency tables of a Markov chain; then we provide an algorithm for drawing contingency tables by using the Markov basis. We also compare the proposed and Dobra's algorithms for generating tables. Section 5 gives MCMC using the Markov basis for generating random tables. The estimates of exact p-values can be calculated from the random tables. In Sect. 6, we present numerical experiments for examining the performance of the proposed MCMC using real datasets and simulation studies. Section 7 gives some concluding remarks.

2 Decomposable log-linear models and junction trees

Let $X = (X_1, \ldots, X_K)$ be a vector of discrete random variables and $\mathcal{V} = \{1, \ldots, K\}$ denote the index set associated with $X_1, \ldots, X_K$. Each variable X_k takes a finite number of values $x_k \in \mathcal{I}_k = \{1, 2, \ldots, I_k\}$ for $k \in \mathcal{V}$. Let $n(x)$ be the entry for cell $x \in \mathcal{I} = \prod_{k \in \mathcal{V}} \mathcal{I}_k$. Then a table of counts $\mathbf{n} = \{n(x)\}_{x \in \mathcal{I}}$ is a K-dimensional array of non-negative integer numbers. For a subset $C \subset \mathcal{V}$, we write X_C for $\{X_k\}_{k \in C}$ and $\mathcal{I}_C = \prod_{k \in C} \mathcal{I}_k$. Then the marginal $\mathbf{n}_C$ of $\mathbf{n}$ is the contingency table with marginal cells $x_C \in \mathcal{I}_C$ and entries given by

$$n_C(x_C) = \sum_{x \in \mathcal{I}_{\mathcal{V} \setminus C}} n(x_C, x),$$

where $(x_C, x) \in \mathcal{I}$.

Consider a hierarchical log-linear model specified by a generating class $\mathcal{C} = \{C_i\}_{1 \leq i \leq M}$ which is a class of variable sets in minimal sufficient statistics. The log-linear model is *graphical* if $\mathcal{C}$ is the set of cliques of an independence graph. Moreover, the log-linear model is *decomposable* if the independence graph induced by $\mathcal{C}$ is a hypergraph $\mathcal{G} = (\mathcal{V}, \mathcal{C})$, where $\mathcal{V}$ is the set of vertices and $\mathcal{C}$ is the set of cliques, i.e., the generating class. Detail discussions of decomposable log-linear models can be found in Lauritzen (1996).

It is known that all the cliques in $\mathcal{C}$ of $\mathcal{G}$ can be ordered to satisfy the *running intersection property*: For all $1 < j \leq M$, there exists an $i < j$ such that

$$C_j \cap (C_1 \cup \cdots \cup C_{j-1}) \subset C_i.$$

By employing the maximum cardinality search algorithm described in Hájek et al. (1992), the ordered clique set $\mathcal{C}$ can be obtained. When the cliques in $\mathcal{C}$ have the running intersection property, it is possible to construct a *junction tree* $\mathcal{T}(\mathcal{C})$ with $\mathcal{C}$ as its node set, see Jensen and Jensen (1994). We define a *clique separator* S_i for $C_i \in \mathcal{C}$ as

$$S_i = C_i \cap \left(\bigcup_{h=1}^{i-1} C_h \right)$$

for $i = 2, \ldots, M$, and denote the set of separators by $\mathcal{S} = \{S_i\}_{2 \leq i \leq M}$. Then a separator $S \in \mathcal{S}$ is associated with the edge between a pair of adjacent cliques in $\mathcal{T}(\mathcal{C})$.

For $\mathcal{T}(\mathcal{C})$, the following lemma is immediately obtained:

Lemma 1 *For any two cliques C_i, C_j $(i < j)$ in $\mathcal{C}$ of $\mathcal{T}(\mathcal{C})$, there exists a unique path from C_i to C_j that is the sequence $C_i = C'_0, C'_1, \ldots, C'_{l-1}, C'_l = C_j$ of distinct cliques.*

We can then characterize the conditional independence relationship between any two cliques C_i, C_j on $\mathcal{T}(\mathcal{C})$ as follows:

Theorem 1 *Assume that there exists a path from C_i to C_j $(i < j)$ such that $C_i = C'_0, C'_1, \ldots, C'_{l-1}, C'_l = C_j$. We set A_i, A_j and B as*

$$A_i = C_i \backslash S'_i, \qquad A_j = C_j \backslash S_j, \qquad B = \mathcal{V} \backslash (A_i \cup A_j), \tag{1}$$

where $S'_i = C_i \cap C'_1 \in \mathcal{S}$. Then X_{A_i} is conditionally independent of X_{A_j} given X_B, i.e., $X_{A_i} \perp X_{A_j} | X_B$.

Proof The global Markov property defined by Lauritzen (1996) leads to the above conditional independence relationship. □

Theorem 1 plays an important role for generating Markov bases in Sect. 3.

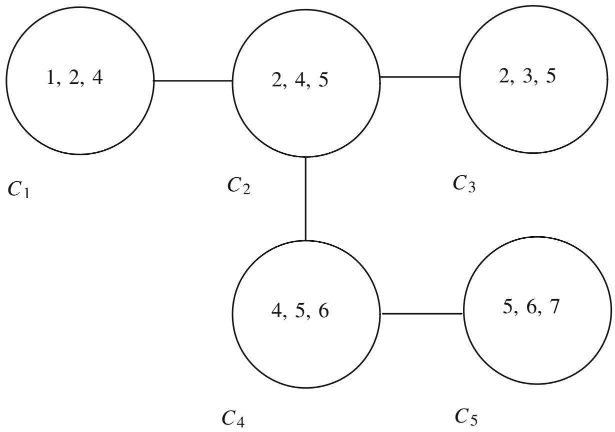

Fig. 1 The junction tree $\mathcal{T}(\mathcal{C})$

Example 2.1 Let $\mathbf{n}$ be a seven-way contingency table of a vector of discrete random variables $X = (X_1, \ldots, X_7)$ and $\mathcal{V} = \{1, \ldots, 7\}$ be the index set of X. Consider the decomposable log-linear model with generating class $\mathcal{C}$ that is the set of ordered cliques such that $C_1 = \{1, 2, 4\}$, $C_2 = \{2, 4, 5\}$, $C_3 = \{2, 3, 5\}$, $C_4 = \{4, 5, 6\}$ and $C_5 = \{5, 6, 7\}$. The marginal tables $\mathbf{n}_{C_1}, \ldots, \mathbf{n}_{C_5}$ are also the minimal sufficient statistics for the log-linear model.

For the hypergraph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ of the log-linear model, the separators in $\mathcal{S}$ are given by $S_2 = \{2, 4\}$, $S_3 = \{2, 5\}$, $S_4 = \{4, 5\}$ and $S_5 = \{5, 6\}$. The junction tree $\mathcal{T}(\mathcal{C})$ defined by $\mathcal{C}$ is illustrated in Fig. 1. For example, the path from C_3 to C_5 in $\mathcal{T}(\mathcal{C})$ is the sequence C_3, C_2, C_4, C_5. Then we have $A_3 = C_3 \backslash S_3 = \{3\}$, $A_5 = C_5 \backslash S_5 = \{7\}$ and $B = \mathcal{V} \backslash (A_2 \cup A_5) = \{1, 2, 4, 5, 6\}$, so that we can obtain $X_{A_3} \perp X_{A_5} | X_B$, or $X_3 \perp X_7 | \{X_1, X_2, X_4, X_5, X_6\}$.

3 Markov bases for decomposable models

Let $\Gamma(\mathcal{C})$ denote the set of all K-way contingency tables of non-negative integer entries with fixed marginals $\mathbf{n}_{C_1}, \ldots, \mathbf{n}_{C_M}$. To generate a table $\mathbf{n}' \in \Gamma(\mathcal{C})$ from an original table $\mathbf{n} \in \Gamma(\mathcal{C})$, we can use the data swapping technique of Dalenius and Reiss (1982) such that cell entries are moved from one cell to the other while the fixed marginals are left unchanged. Since some of the cell entries are increased and other cells are decreased in the swapping, a *data swap* associated with $\mathbf{n}$ is an array $\mathbf{f} = \{f(x)\}_{x \in \mathcal{I}}$ with integer entries $f(x) \in \{0, \pm 1, \pm 2, \ldots\}$. Thus $\mathbf{n}'$ created by adding $\mathbf{f}$ to $\mathbf{n}$ has to be consistent with fixed marginals.

Definition 1 A move $\mathbf{f}$ for $\mathcal{C}$ is a data swap that preserves fixed marginals $\mathbf{n}_{C_1}, \ldots, \mathbf{n}_{C_M}$ specified by $\mathcal{C}$.

Then we have $\mathbf{n} + \mathbf{f} \in \Gamma(\mathcal{C})$ if and only if $n(x) + f(x) \geq 0$ for all $x \in \mathcal{I}$. From the marginal constraints associated with clique $C \in \mathcal{C}$, we have

$$\sum_{x \in \mathcal{I}_{\mathcal{V} \setminus C}} n(x_C, x) + f(x_C, x) = n_C(x_C) \quad \text{and}$$
$$\sum_{x \in \mathcal{I}} f(x) = \sum_{x_C \in \mathcal{I}_C} \sum_{x \in \mathcal{I}_{\mathcal{V} \setminus C}} f(x_C, x) = 0.$$

The move with entries of zero everywhere except four cells that contain two entries of 1 and two entries of -1 is called a *primitive move*.

Definition 2 A Markov basis $\mathcal{M}$ is a finite collection of moves that preserve fixed marginals $\mathbf{n}_{C_1}, \ldots, \mathbf{n}_{C_M}$. For any two tables $\mathbf{n}, \mathbf{n}' \in \Gamma(\mathcal{C})$, there exists a sequence of moves $\mathbf{f}^{(1)}, \ldots, \mathbf{f}^{(R)} \in \mathcal{M}$ such that

$$\mathbf{n}' - \mathbf{n} = \sum_{r=1}^{R} \mathbf{f}^{(r)} \quad \text{and} \quad \mathbf{n} + \sum_{r=1}^{R'} \mathbf{f}^{(r)} \in \Gamma(\mathcal{C}),$$

for $1 \leq R' \leq R$.

From the result of Diaconis and Sturmfels (1998), there exists a Markov basis for any $\Gamma(\mathcal{C})$. The Markov basis allows the construction of an irreducible Markov chain on $\Gamma(\mathcal{C})$. Diaconis and Sturmfels (1998) also described a Markov basis of primitive moves for the independence model for a two-way contingency table that is the simplest decomposable log-linear model.

We extend their idea for generating primitive moves for the independence model for a two-way table to the case of an arbitrary decomposable log-linear model of a multi-way table. Let $\mathcal{C} = \{C_1, \ldots, C_M\}$ be the set of ordered cliques of a decomposable graph $\mathcal{G}$ or a generating class of a decomposable log-linear model. We describe a Markov basis for $\Gamma(C_i, C_j)$ that is the set of all contingency tables with fixed marginals $\mathbf{n}_{C_i}$ and $\mathbf{n}_{C_j}$. For any two cliques $C_i, C_j (i < j)$ in $\mathcal{C}$, the decomposable log-linear model has the conditional independence structure $X_{A_i} \perp X_{A_j} | X_B$ from Eq. (1). We break up $\mathbf{n}$ into slices corresponding to each value of X_B and obtain x_B-slice of the table, $\{\mathbf{n}^{x_B}_{A_i \cup A_j}\}_{x_B \in \mathcal{I}_B}$, where $\mathbf{n}^{x_B}_{A_i \cup A_j} = \{n(x_{A_i}, x_{A_j}, x_B)\}_{(x_{A_i}, x_{A_j}) \in \mathcal{I}_{A_i} \times \mathcal{I}_{A_j}}$ is cross-classified by X_{A_i} and X_{A_j} for fixed $x_B \in \mathcal{I}_B$. Note that $\{\mathbf{n}^{x_B}_{A_i \cup A_j}\}_{x_B \in \mathcal{I}_B}$ preserves fixed marginals $\mathbf{n}_{C_1}, \ldots, \mathbf{n}_{C_M}$ and thus $\{\mathbf{n}^{x_B}_{A_i \cup A_j}\}_{x_B \in \mathcal{I}_B} \in \Gamma(\mathcal{C})$. For some indices $x^1_{A_i} \neq x^2_{A_i} \in \mathcal{I}_{A_i}$, $x^1_{A_j} \neq x^2_{A_j} \in \mathcal{I}_{A_j}$, we define a primitive move associated with $\mathbf{n}^{x_B}_{A_i \cup A_j}$ by $\mathbf{f}^{x_B}_{A_i \cup A_j} = \{f(x_{A_i}, x_{A_j}, x_B)\}_{(x_{A_i}, x_{A_j}) \in \mathcal{I}_{A_i} \times \mathcal{I}_{A_j}}$, where

$$f(x_{A_i}, x_{A_j}, x_B) = \begin{cases} 1, & \text{if } (x_{A_i}, x_{A_j}) \in \{(x^1_{A_i}, x^1_{A_j}), (x^2_{A_i}, x^2_{A_j})\} \quad \text{with } x_B \text{ fixed}, \\ -1, & \text{if } (x_{A_i}, x_{A_j}) \in \{(x^1_{A_i}, x^2_{A_j}), (x^2_{A_i}, x^1_{A_j})\} \quad \text{with } x_B \text{ fixed}, \\ 0, & \text{otherwise}. \end{cases}$$

Then

$$\mathbf{f}^{x_B} = \{\pm \mathbf{f}^{x_B}_{A_i \cup A_j}\}_{(x^1_{A_i}, x^2_{A_i}) \in \mathcal{I}_{A_i} \times \mathcal{I}_{A_i}, (x^1_{A_j}, x^2_{A_j}) \in \mathcal{I}_{A_j} \times \mathcal{I}_{A_j}} \tag{2}$$

is a Markov basis for $\mathbf{n}^{x_B}_{A_i \cup A_j}$.

The following theorem shows a Markov basis for $\Gamma(C_i, C_j)$.

Theorem 2 *Let C_i and C_j be any two clique in $\mathcal{C}$. Then*

$$\mathcal{F}(C_i, C_j) = \sum_{x_B \in \mathcal{I}_B} \mathbf{f}^{x_B} \tag{3}$$

is a Markov basis for $\Gamma(C_i, C_j)$.

Proof Let $\mathbf{f} \in \mathcal{F}(C_i, C_j)$. We have

$$\sum_{x \in \mathcal{I}_{A_j} \times \mathcal{I}_{B \setminus C_i}} n(x_{C_i}, x) + f(x_{C_i}, x) = n_{C_i}(x_{C_i}) \quad \text{for all } x_{C_i} \in \mathcal{I}_{C_i},$$

$$\sum_{x \in \mathcal{I}_{A_i} \times \mathcal{I}_{B \setminus C_j}} n(x_{C_j}, x) + f(x_{C_j}, x) = n_{C_j}(x_{C_j}) \quad \text{for all } x_{C_j} \in \mathcal{I}_{C_j}.$$

Thus $\mathbf{n} + \mathbf{f} \in \Gamma(C_i, C_j)$.

Next we prove that $\mathcal{F}(C_i, C_j)$ is a Markov basis for $\Gamma(C_i, C_j)$. From Theorem 1, we have $X_{A_i} \perp X_{A_j} | X_B$. We recast X_{A_i}, X_{A_j} and X_B by new variables Y_1, Y_2 and Y_3 with $I_1 = \prod_{k \in A_i} I_k$, $I_2 = \prod_{k \in A_j} I_k$ and $I_3 = \prod_{k \in B} I_k$ categories defined by $\mathcal{I}_1$, $\mathcal{I}_2$ and $\mathcal{I}_3$, respectively. We then have $Y_1 \perp Y_2 | Y_3$. Thus it is sufficient to give a Markov basis for the log-linear model with the generating class $\{C_1, C_2\}$ such that $C_1 = \{1, 3\}$ and $C_2 = \{2, 3\}$.

Let $\mathbf{n}$ be cell counts of the $I_1 \times I_2 \times I_3$ table. By considering the reverse lexicographic order of the cell indices in $\mathcal{I}_1 \times \mathcal{I}_2 \times \mathcal{I}_3$, $\mathbf{n}$ can be transformed into a linear list of counts (i.e. vector) $\bar{\mathbf{n}}$. Given a log-linear model, the sufficient statistics $\mathbf{b} = (\bar{\mathbf{n}}_{C_1}, \bar{\mathbf{n}}_{C_2})$ with the reverse lexicographic order of the cell indices can be written as $A\bar{\mathbf{n}} = \mathbf{b}$, where A is a some integer matrix and called a configuration of the model. For the log-linear model with $\{C_1, C_2\} = \{\{1, 3\}, \{2, 3\}\}$, A is given by

$$A = \begin{pmatrix} E_{I_3} \otimes \mathbf{1}_{I_2}^\top \otimes E_{I_1} \\ E_{I_3} \otimes E_{I_2} \otimes \mathbf{1}_{I_1}^\top \end{pmatrix},$$

where E_I is the $I \times I$ identity matrix, $\mathbf{1}_I$ is the column vector of ones length I, and $\otimes$ denotes the Kronecker product. After permuting rows of A, we have

$$A' = \begin{pmatrix} \mathbf{e}_1 \otimes \mathbf{1}_{I_2}^\top \otimes E_{I_1} \\ \mathbf{e}_1 \otimes E_{I_2} \otimes \mathbf{1}_{I_1}^\top \\ \vdots \\ \mathbf{e}_{I_3} \otimes \mathbf{1}_{I_2}^\top \otimes E_{I_1} \\ \mathbf{e}_{I_3} \otimes E_{I_2} \otimes \mathbf{1}_{I_1}^\top \end{pmatrix},$$

where $\mathbf{e}_k$ is the k-th row of E_{I_3}. By denoting

$$A'_k = \begin{pmatrix} \mathbf{e}_k \otimes \mathbf{1}_{I_2}^\top \otimes E_{I_1} \\ \mathbf{e}_k \otimes E_{I_2} \otimes \mathbf{1}_{I_1}^\top \end{pmatrix} = (\underbrace{0_{I_1 I_2} \ \ \ldots \ \ 0_{I_1 I_2}}_{k-1} \ \ A_{I_1 I_2} \ \ \underbrace{0_{I_1 I_2} \ \ \ldots \ \ 0_{I_1 I_2}}_{I_3 - k}),$$

A' is written by the block diagonal matrix

$$A' = \begin{pmatrix} A'_1 \\ A'_2 \\ \vdots \\ A'_{I_3} \end{pmatrix} = \begin{pmatrix} A_{I_1 I_2} & & & 0 \\ & A_{I_1 I_2} & & \\ & & \ddots & \\ 0 & & & A_{I_1 I_2} \end{pmatrix},$$

where $A_{I_1 I_2}$ is the $(I_1 + I_2) \times I_1 I_2$ matrix given by

$$A_{I_1 I_2} = \begin{pmatrix} \mathbf{1}_{I_2}^\top \otimes E_{I_1} \\ E_{I_2} \otimes \mathbf{1}_{I_1}^\top \end{pmatrix}$$

and $0_{I_1 I_2}$ is the $(I_1 + I_2) \times I_1 I_2$ matrix of all zero elements. Note that $A_{I_1 I_2}$ is the same form of the configuration of the independence model for a two-way contingency table.

Because A' is the block diagonal matrix, we can compute independently the Gröbner basis for the toric ideal associated with A'_k and then obtain a Markov basis $\mathbf{f}^k$ of Eq. (2). Therefore, $\mathcal{F}(C_1, C_2)$ of Eq. (3) given by the linear combination

$$\mathbf{f} = \sum_{k \in \mathcal{I}_3} \mathbf{f}^k$$

of Markov bases $\{\mathbf{f}^k\}_{k \in \mathcal{I}_3}$ is indeed a Markov basis for $\Gamma(C_1, C_2)$. □

In "Appendix", we give A' of the conditional independence model of a three-way contingency table.

Using Eq. (3), we define a Markov basis on the junction tree for a decomposable log-linear model as follows:

Definition 3 Let $\mathcal{C} = \{C_1, \ldots, C_M\}$ be the set of ordered cliques of a decomposable graph $\mathcal{G}$ and let $\mathcal{T}(\mathcal{C})$ be the junction tree with $\mathcal{C}$ as its node set. Then the set of primitive moves associated with the junction tree $\mathcal{T}(\mathcal{C})$ is defined by

$$\mathcal{F}(\mathcal{T}(\mathcal{C})) = \mathcal{F}(C_1, \ldots, C_M) = \bigcup_{C_i, C_j (\neq C_i) \in \mathcal{C}} \mathcal{F}(C_i, C_j), \tag{4}$$

where $\mathcal{F}(C_i, C_j)$ is given by Eq. (3).

We provide the following theorem for a Markov basis for $\Gamma(\mathcal{C})$.

Theorem 3 *Let $\mathcal{T}(\mathcal{C})$ be the junction tree with $\mathcal{C} = \{C_1, \ldots, C_M\}$ as its node set. Then the set of primitive moves $\mathcal{F}(\mathcal{T}(\mathcal{C})) = \mathcal{F}(C_1, \ldots, C_M)$ defined by Eq. (4) is a set of moves for $\Gamma(\mathcal{C})$. Furthermore $\mathcal{F}(\mathcal{T}(\mathcal{C}))$ is a Markov basis for $\Gamma(\mathcal{C})$.*

Proof By Definition 3, we have from Theorem 2 that $\mathcal{F}(\mathcal{T}(\mathcal{C}))$ is a Markov basis for $\Gamma(\mathcal{C})$. Thus the result can be obtained immediately. □

4 Generating Markov bases on junction trees

4.1 Algorithm for generating random tables

We propose an algorithm for generating a random table in $\Gamma(\mathcal{C})$. Assume that the junction tree $\mathcal{T}(\mathcal{C})$ of the ordered clique set $\mathcal{C}$ can be constructed. We calculate weight w_{ij} representing the number of moves for $\mathcal{F}(C_i, C_j)$ by

$$w_{ij} = \left\{ 2 \times \binom{I_{A_i}}{2} \times \binom{I_{A_j}}{2} \right\}^{I_B},$$

where $I_{A_i} = \prod_{k \in A_i} I_k$, $I_{A_j} = \prod_{k \in A_j} I_k$ and $I_B = \prod_{k \in B} I_k$, and normalize w_{ij} by $p(w_{ij}) = w_{ij} / \sum_{1 \le s < t \le M} w_{st}$.

Given a contingency table $\mathbf{n}$, the following algorithm generates a candidate table $\tilde{\mathbf{n}}$ by using $\mathcal{F}(\mathcal{T}(\mathcal{C}))$:

[**Algorithm 1: Random table generation**]

Step 1: Randomly select two cliques C_i, C_j $(i < j)$ in $\mathcal{C}$ with probability $p(w_{ij})$. Obtain A_i, A_j and B by

$$A_i = \begin{cases} C_i \setminus S_j, & \text{if } C_i \cap C_j = S_j, \\ C_i \setminus \left(\bigcup_{h=i+1}^{j} S_h \right), & \text{otherwise,} \end{cases}$$
$$A_j = C_j \setminus S_j,$$
$$B = \mathcal{V} \setminus (A_i \cup A_j).$$

Step 2: Uniformly choose a move $\mathbf{f} \in \mathcal{F}(C_i, C_j)$.
Step 3: Generate $\tilde{\mathbf{n}}$ by $\tilde{\mathbf{n}} = \mathbf{n} + \mathbf{f}$.

Then **Random table generation** produces $\tilde{\mathbf{n}}$ that $4I_B$ cell entries in $\mathbf{n}$ are changed.

Example 4.1 We illustrate **Random table generation** by using the junction tree introduced in Example 2.1. Assume that C_3 and C_5 are selected with probability $p(w_{35})$ in Step 1. From Eq. (1), we have $A_3 = \{3\}$, $A_5 = \{7\}$ and $B = \{1, 2, 4, 5, 6\}$, i.e., $X_{A_3} = X_3$, $X_{A_5} = X_7$ and $X_B = (X_1, X_2, X_4, X_5, X_6)$. Step 2 uniformly selects a move $\mathbf{f} \in \mathcal{F}(C_3, C_5)$. To pick $\mathbf{f}$, the following steps are repeated for every $x_B \in \mathcal{I}_B$:

- Choose indices $x_3^1 \neq x_3^2 \in \mathcal{I}_3$, $x_7^1 \neq x_7^2 \in \mathcal{I}_7$.
- Obtain a move $\mathbf{f}_{\{3,7\}}^{x_B} = \{f(x_3, x_7, x_B)\}$, where

$$f(x_3, x_7, x_B) = \begin{cases} 1, & \text{if } (x_3, x_7) \in \{(x_3^1, x_7^1), (x_3^2, x_7^2)\} \text{ with fixed } x_B, \\ -1, & \text{if } (x_3, x_7) \in \{(x_3^1, x_7^2), (x_3^2, x_7^1)\} \text{ with fixed } x_B, \\ 0, & \text{otherwise.} \end{cases}$$

We compute $\mathbf{f} = \sum_{x_B \in \mathcal{I}_B} \varepsilon \mathbf{f}_{\{3,7\}}^{x_B}$, where ε is -1 or $+1$ with probability $1/2$ each. Step 3 generates $\tilde{\mathbf{n}}$ by $\tilde{\mathbf{n}} = \mathbf{n} + \mathbf{f}$.

4.2 Comparison of the proposed and Dobra's algorithms for generating random tables

Dobra (2003) defined a Markov basis that is the set of primitive moves and is characterized by the star property of Blair and Barry (1993), which decomposes $\mathcal{G}$ into the triplet of disjoint subsets $(\mathcal{V}_1, S, \mathcal{V}_2)$ such that $X_{\mathcal{V}_1} \perp X_{\mathcal{V}_2} | X_S$, where $S \in \mathcal{S}$. When $\mathcal{T}(\mathcal{C})$ has the star property such that S separates $\mathcal{V}_1$ from $\mathcal{V}_2$ in $\mathcal{G}$, it is possible to construct $\mathcal{T}(\{\mathcal{V}_1 \cup S, \mathcal{V}_2 \cup S\})$. When decomposing $\mathcal{G}$ using the global Markov property in Theorem 1, we have $\mathcal{T}(\{\mathcal{A}_i \cup B, \mathcal{A}_j \cup B\})$.

For a given table $\mathbf{n}$, Dobra's random table generation algorithm based on the star property produces a candidate table $\tilde{\mathbf{n}}$ that 4 cell entries in $\mathbf{n}$ are changed. We can see that the number of cells changed by **Random table generation** is I_B times of that by Dobra's algorithm, and thus **Random table generation** produces a variety of contingency tables in the set of contingency tables with fixed marginals.

Example 4.2 We show the number of cell entries that are changed by using both algorithms. We assume that **Random table generation** selects C_2 and C_4 on the junction tree introduced in Example 2.1. Then we have $A_2 = \{2\}$, $A_4 = \{6\}$ and $B = \{1, 3, 4, 5, 7\}$ by the decomposition of the global Markov property.

For given (A_2, A_4, B), we break up $\mathbf{n}$ into $\{\mathbf{n}^{x_B}_{\{2,6\}}\}_{x_B \in \mathcal{I}_B}$. In the case of all variables of X being binary, we have 32 x_B-slices of the 2×2 table of X_2 and X_6. We denote $X_B = (X_1, X_3, X_4, X_5, X_7) = (1, 1, 1, 1, 1) = \mathbf{1}$, $X_B = (1, 1, 1, 1, 2) = \mathbf{2}$, ..., $X_B = (2, 2, 2, 2, 2) = \mathbf{32}$. Then $\{\mathbf{n}^{x_B}_{\{2,6\}}\}_{x_B \in \{\mathbf{1},\ldots,\mathbf{32}\}}$ is displayed as follows:

$$\begin{array}{ccc}
 & & \begin{array}{cc} X_6 = 1 & X_6 = 2 \end{array} \\
X_B = \mathbf{1} & \begin{array}{c} X_2 = 1 \\ X_2 = 2 \end{array} & \boxed{\begin{array}{cc} n(1,1,\mathbf{1}) & n(1,2,\mathbf{1}) \\ n(2,1,\mathbf{1}) & n(2,2,\mathbf{1}) \end{array}} \\
 & & \\
 & & \begin{array}{cc} X_6 = 1 & X_6 = 2 \end{array} \\
X_B = \mathbf{2} & \begin{array}{c} X_2 = 1 \\ X_2 = 2 \end{array} & \boxed{\begin{array}{cc} n(1,1,\mathbf{2}) & n(1,2,\mathbf{2}) \\ n(2,1,\mathbf{2}) & n(2,2,\mathbf{2}) \end{array}} \\
\vdots & & \vdots \\
 & & \begin{array}{cc} X_6 = 1 & X_6 = 2 \end{array} \\
X_B = \mathbf{32} & \begin{array}{c} X_2 = 1 \\ X_2 = 2 \end{array} & \boxed{\begin{array}{cc} n(1,1,\mathbf{32}) & n(1,2,\mathbf{32}) \\ n(2,1,\mathbf{32}) & n(2,2,\mathbf{32}) \end{array}}
\end{array}$$

For each x_B, **Random table generation** produces $\tilde{\mathbf{n}}^{x_B}_{\{2,6\}} = \mathbf{n}^{x_B}_{\{2,6\}} + \varepsilon \mathbf{f}^{x_B}_{\{2,6\}}$, where $\mathbf{f}^{x_B}_{\{2,6\}} \in \{-1, 1\}$. Thus **Random table generation** enables changing all $128 (= 4 \times 32)$ cell entries in $\tilde{\mathbf{n}}$ whereas only 4 cell entries in $\tilde{\mathbf{n}}$ are changed by using Dobra's algorithm.

5 Computing *p*-values using MCMC with Markov bases

We provide MCMC using $\mathcal{F}(\mathcal{T}(\mathcal{C}))$ for generating contingency tables of a Markov chain on $\Gamma(\mathcal{C})$. Then the Metropolis–Hastings (M–H) algorithm of Hastings (1970) can be used to generate the Markov chain. Diaconis and Sturmfels (1998) proved that the Markov chain generated by the M–H algorithm is an irreducible, aperiodic Markov chain on $\Gamma(\mathcal{C})$ with a stationary distribution π.

5.1 Generating Markov chains in MCMC

We assume that $\mathbf{n}$ has a multinomial distribution. The exact conditional distribution of $\mathbf{n}$ given the set of marginal constraints determined by $X_{A_i} \perp X_{A_j} | X_B$ has the product hypergeometric distribution with the following probability function:

$$\begin{aligned}\pi(\mathbf{n}) &= \prod_{x_B \in \mathcal{I}_B} \pi\left(\mathbf{n}^{x_B}_{A_i \cup A_j}\right) \\ &= \prod_{x_B \in \mathcal{I}_B} \left\{ \frac{\prod_{x_{A_i} \in \mathcal{I}_{A_i}} n(x_{A_i}, x_B)! \prod_{x_{A_j} \in \mathcal{I}_{A_j}} n(x_{A_j}, x_B)!}{n(x_B)! \prod_{(x_{A_i}, x_{A_j}) \in \mathcal{I}_{A_i} \times \mathcal{I}_{A_j}} n(x_{A_i}, x_{A_j}, x_B)!} \right\}. \end{aligned} \tag{5}$$

Equation (5) indicates that there exists mutual independence between tables $\{\mathbf{n}^{x_B}_{A_i \cup A_j}\}_{x_B \in \mathcal{I}_B}$ conditional on the fixed marginal table $\mathbf{n}_B$.

To generate random tables of a Markov chain on $\Gamma(\mathcal{C})$, the M–H algorithm iterates through the following steps:

[Algorithm 2: M-H algorithm for generating a Markov chain]

Step 0: Initialize the iteration counter $r = 1$ and set $\mathbf{n}^{(0)}$ as the initial contingency table.

Step 1: Generate a candidate table $\tilde{\mathbf{n}} = \{\tilde{\mathbf{n}}^{x_B}_{A_i \cup A_j}\}_{x_B \in \mathcal{I}_B}$ by using **Random table generation**.

Step 2: For every $x_B \in \mathcal{I}_B$, proceed as follows:

- If all cell entries of $\tilde{\mathbf{n}}^{x_B}_{A_i \cup A_j}$ are non-negative integers, accept $\tilde{\mathbf{n}}^{x_B}_{A_i \cup A_j}$ as the next table $\mathbf{n}^{x_B\ (r)}_{A_i \cup A_j}$ with probability

$$\alpha(\tilde{\mathbf{n}}^{x_B}_{A_i \cup A_j}, \mathbf{n}^{x_B\ (r-1)}_{A_i \cup A_j}) = \min\left\{ \frac{\pi(\tilde{\mathbf{n}}^{x_B}_{A_i \cup A_j})}{\pi(\mathbf{n}^{x_B\ (r-1)}_{A_i \cup A_j})}, 1 \right\}, \tag{6}$$

otherwise reject $\tilde{\mathbf{n}}^{x_B}_{A_i \cup A_j}$ and set $\mathbf{n}^{x_B\ (r)}_{A_i \cup A_j} = \mathbf{n}^{x_B\ (r-1)}_{A_i \cup A_j}$.

Obtain $\mathbf{n}^{(r)} = \{\mathbf{n}^{x_B\ (r)}_{A_i \cup A_j}\}_{x_B \in \mathcal{I}_B} \in \Gamma(\mathcal{C})$.

Step 3: Increment counter r and return to Step 1.

5.2 MCMC computation of p-values and their standard errors

The M–H algorithm draws $\{\mathbf{n}^{(r)}\}_{0 \le r \le R_0 + R} \in \Gamma(\mathcal{C})$ for a specified clique set $\mathcal{C}$. Then, after discarding the first R_0 tables as a burn-in, the p-value via MCMC (MCMC p-value) can be obtained by

$$\hat{p} = \frac{1}{R} \sum_{r=R_0+1}^{R_0+R} I\left\{\chi^2(\mathbf{n}^{(r)}, \mathbf{m}) \geq \chi^2(\mathbf{n}, \mathbf{m})\right\}, \tag{7}$$

where $\chi^2(\mathbf{n}, \mathbf{m})$ is the Pearson chi-squared statistic given by

$$\chi^2(\mathbf{n}, \mathbf{m}) = \sum_{x \in \mathcal{I}} \frac{\{n(x) - m(x)\}^2}{m(x)}$$

and $\mathbf{m} = \{m(x)\}_{x \in \mathcal{I}}$ is the set of expected values conditional on marginal constrains of $\mathcal{C}$.

To evaluate the accuracy of the MCMC estimates, we compute the standard error of $\hat{p}$ using the batch means method of Geyer (1992), which works as follows: Dividing random tables $\{\mathbf{n}^{(r)}\}_{R_0+1 \leq r \leq R_0+R_1R_2}$ into R_1 batches of size R_2 each, the MCMC p-value for the kth batch is then calculated by

$$\hat{p}_k = \frac{1}{R_2} \sum_{r=R_0+(k-1)R_2+1}^{R_0+kR_2} I\left\{\chi^2(\mathbf{n}^{(r)}, \mathbf{m}) \geq \chi^2(\mathbf{n}, \mathbf{m})\right\}.$$

The variance estimate of $\hat{p}$ can be obtained from

$$\mathrm{Var} = \frac{1}{R_1 - 1} \sum_{k=1}^{R_1} (\hat{p}_k - \hat{p})^2 \tag{8}$$

and the batch means estimate of the Monte Carlo standard error is calculated from $\sqrt{\mathrm{Var}/R_1}$. Then the batch size R_1 should be large enough so that $\hat{p}_k$s are approximately independent. Jones et al. (2006) recommended to use $R_1 = \lfloor \sqrt{R} \rfloor$ and $R_2 = \lfloor R/R_1 \rfloor$.

6 Numerical experiments

In this section, we illustrate the performance of the proposed MCMC in comparison with that of asymptotic chi-squared approximation and Dobra's MCMC by using real datasets, and evaluate MCMC for testing decomposable log-linear models via simulation studies.

6.1 MCMC for computing p-values

In the first experiment, we compute the MCMC p-values for testing decomposable log-linear models for sparse contingency tables. The M–H algorithm generates 1,000,000 ($= R$) random tables with 10,000 ($= R_0$) tables as a burn-in. We also set $R_1 = \sqrt{R} = 1,000$ and $R_2 = R/R_1 = 1,000$ for estimating the standard errors of the MCMC p values.

The efficiency of the M–H algorithm is measured by the *inefficiency factor* defined by Var/σ^2, where Var is computed from Eq. (8) and σ^2 is the variance of p-values obtained from hypothetical independent samples. The inefficiency factor is related

Table 1 Study of nonmetastatic osteosarcoma by Goorin et al. (1987)

Sex (X_1)	Lymphocytic infiltration (X_2)	Osteoblastic pathology (X_3)	Disease-free (X_4)	
			Yes	No
Female	High	No	3	0
		Yes	4	0
	Low	No	5	0
		Yes	5	4
Male	High	No	2	0
		Yes	1	0
	Low	No	3	2
		Yes	6	11

to the number of generating samples required to achieve a given degree of numerical accuracy. Thus the algorithms have higher inefficiency factors when they find p-values with larger standard errors.

Experiment 1. Table 1 summarizes the data from a study of nonmetastatic osteosarcoma by Goorin et al. (1987) in which the response is whether the subject achieved a three-year disease-free interval. This table is a sparse table with some cells having zero entries. For the contingency table, we consider the decomposable log-linear models with the junction trees specified by clique sets $\mathcal{C}_1$ to $\mathcal{C}_8$ given in the first column in Table 2.

Table 2 lists the results of the experiments. We can find large discrepancies between the asymptotic p-values obtained using the chi-squared distribution and the MCMC p-values. They indicate that the asymptotic chi-squared approximation is unreliable for sparse contingency tables.

Figure 2 shows the autocorrelation plots for the chain values $\{\chi^2(\mathbf{n}^{(r)}, \mathbf{m})\}_{1 \le r \le R_0}$ of the burn-in generated by the M–H algorithm using **Random table generation** (the proposed M–H algorithm) and the M–H algorithm using Dobra's random table generation algorithm (Dobra's M–H algorithm). We can see from Fig. 2 that the Markov chain generated by the proposed M–H algorithm converges to a stationary state faster than that generated by Dobra's M–H algorithm, because no appreciable dependence exists after lag 20–30 in the Markov chain from the proposed M–H algorithm. We can obtain the same results for the rest models with $\mathcal{C}_5$ to $\mathcal{C}_8$.

We compare the efficiency of two algorithms using the inefficiency factor. Because σ^2 for the two algorithms is the same for each model, we compare their standard errors. For all models, the standard errors of the MCMC p-values by the proposed M–H algorithm are about a half of those of Dobra's M–H algorithm. We conclude that the proposed M–H algorithm is more efficient than Dobra's M–H algorithm for the experiments.

Experiment 2. The second data shown in Table 3 are from the Avadex data given in Innes et al. (1969). We consider the decomposable log-linear models with the junction trees specified by clique sets $\mathcal{C}_1$ to $\mathcal{C}_6$ given in the first column in Table 4.

Table 2 Asymptotic p-values versus MCMC p-values

Clique set	χ^2 value (df)	Asymptotic p-value	MCMC p-value	
			Proposed M–H	Dobra's M–H
$\mathcal{C}_1 = \{\{1,2\},\{2,3\},\{2,4\}\}$	9.267 (8)	0.32024	0.10241	0.10197
			(0.00079)	(0.00162)
$\mathcal{C}_2 = \{\{1,3\},\{2,3\},\{3,4\}\}$	17.846 (8)	0.02241	0.01789	0.01729
			(0.00028)	(0.00054)
$\mathcal{C}_3 = \{\{1,4\},\{2,4\},\{3,4\}\}$	2.655 (8)	0.95410	0.82790	0.83033
			(0.00072)	(0.00141)
$\mathcal{C}_4 = \{\{1,2\},\{1,3\},\{1,4\}\}$	16.187 (8)	0.03978	0.03306	0.03394
			(0.00039)	(0.00081)
$\mathcal{C}_5 = \{\{1,3\},\{3,2\},\{2,4\}\}$	10.332 (8)	0.24250	0.06793	0.06767
			(0.00068)	(0.00138)
$\mathcal{C}_6 = \{\{1,4\},\{2,4\},\{2,3\}\}$	6.116 (8)	0.63424	0.32208	0.31757
			(0.00116)	(0.00242)
$\mathcal{C}_7 = \{\{1,2\},\{1,3\},\{3,4\}\}$	12.833 (8)	0.11772	0.12823	0.12823
			(0.00075)	(0.00148)
$\mathcal{C}_8 = \{\{1,2\},\{2,3\},\{3,4\}\}$	12.250 (8)	0.14040	0.13063	0.13102
			(0.00082)	(0.00167)

The values in parenthesis are the standard errors of the MCMC p-values

The values of the fourth and fifth columns in Table 4 indicate that the MCMC p-values obtained by the proposed M–H algorithm are very close to those by Dobra's M–H algorithm. The standard errors of the MCMC p-values obtained by the proposed M–H algorithm are two times smaller than those by Dobra's M–H algorithm, and thus the proposed M–H algorithm is two times efficient than Dobra's M–H algorithm in terms of the inefficiency factor.

Experiment 3. The third data shown in Table 5 are from a study of the presence or absence of torus mandibularis in three Aleutian populations from Muller and Mayhall (1971). For the table, we consider the decomposable log-linear models with the junction trees specified by clique sets $\mathcal{C}_1$ to $\mathcal{C}_6$ given in the first column in Table 6.

We see from the fourth and fifth columns in Table 6 that the standard errors found by the proposed M–H algorithm are 1.3–1.6 and 2.4 times smaller than those by Dobra's algorithm except the result for $\mathcal{C}_4$. These results show that the proposed M–H algorithm efficiently generates random tables more than Dobra's M–H algorithm. The asymptotic chi-squared approximation seems to work well as it returns asymptotic p-values in agreement with the MCMC p-values.

6.2 MCMC for testing decomposable log-linear models

In the experiment, we focus on the simulation studies of MCMC for testing decomposable log-linear models. The simulation design is as follows: We draw $\mathbf{n}$ from the

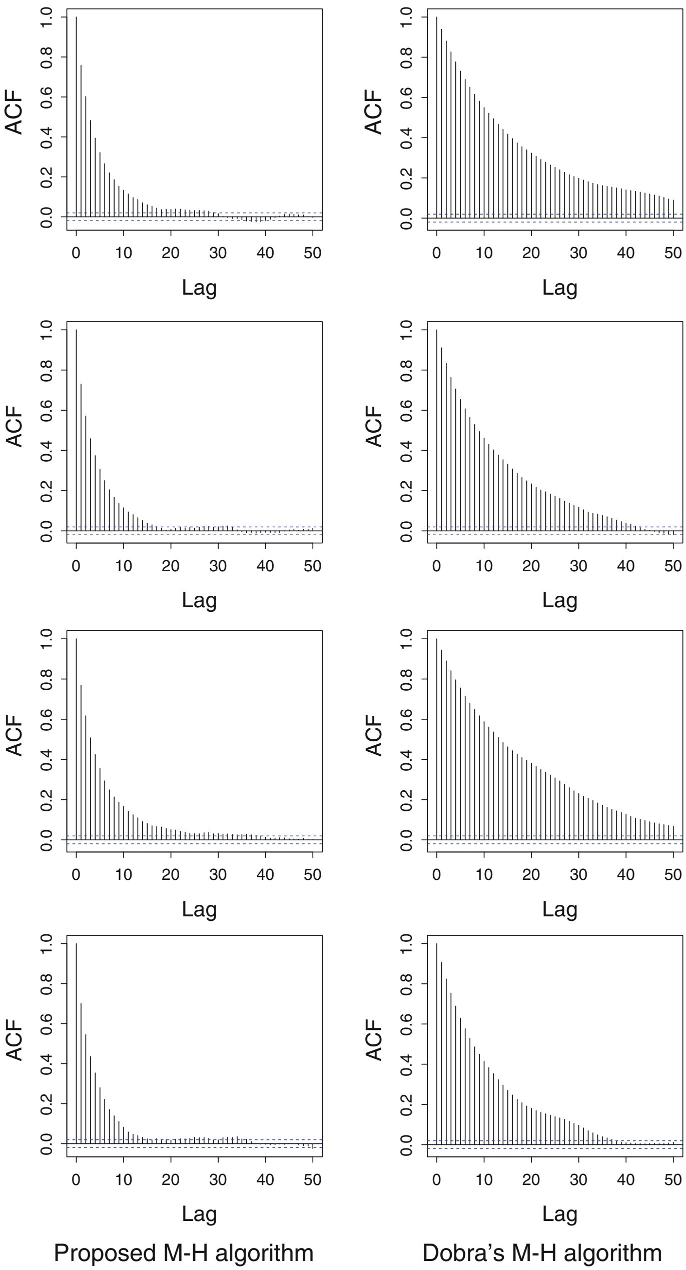

Fig. 2 Autocorrelation plots of the chi-squared statistic by the proposed and Dobra's M–H algorithms (the decomposable log-linear models with the junction trees specified by $\mathcal{C}_1$ to $\mathcal{C}_4$ given in Table 2 are arranged from *top* to *bottom*).

multinomial distribution with sample size of 50 and cell probabilities conditional on $\mathcal{C}$ as given in Table 7. We compute the asymptotic and MCMC p-values for testing decomposable log-linear models with the junction trees specified by the clique sets

Table 3 Lung cancer in mice exposed to Avadex from Innes et al. (1969)

Strain (X_1)	Sex (X_2)	Exposure status (X_3)	Tumors (X_4) Y	N
X	Male	Exposed	12	4
		Control	74	5
	Female	Exposed	12	2
		Control	84	3
Y	Male	Exposed	14	4
		Control	80	10
	Female	Exposed	14	1
		Control	79	3

Table 4 Asymptotic p-values versus MCMC p-values

Clique set	χ^2 value (df)	Asymptotic p-value	MCMC p-value Proposed M–H	Dobra's M–H
$\mathcal{C}_1 = \{\{1,2\},\{2,3\},\{3,4\}\}$	8.164 (8)	0.41757	0.42449	0.42619
			(0.00176)	(0.00338)
$\mathcal{C}_2 = \{\{1,3\},\{2,3\},\{2,4\}\}$	11.385 (8)	0.18085	0.17664	0.17851
			(0.00151)	(0.00314)
$\mathcal{C}_3 = \{\{1,4\},\{2,4\},\{2,3\}\}$	11.609 (8)	0.16950	0.16624	0.16586
			(0.00157)	(0.00286)
$\mathcal{C}_4 = \{\{1,4\},\{2,3\},\{3,4\}\}$	8.185 (8)	0.41562	0.42575	0.42149
			(0.00200)	(0.00374)
$\mathcal{C}_5 = \{\{1,2\},\{2,3\},\{2,4\}\}$	10.632 (8)	0.22346	0.21667	0.21324
			(0.00153)	(0.00293)
$\mathcal{C}_6 = \{\{1,3\},\{2,3\},\{3,4\}\}$	8.925 (8)	0.34867	0.35577	0.35226
			(0.00194)	(0.00384)

The values in parenthesis are the standard errors of the MCMC p-values

given in Table 7. Here the proposed M–H algorithm generates 1,000 random tables and the MCMC p-values are obtained from Eq. (7). The above procedure is replicated 1,000 times. Then the Type I error rate is computed from the proportion of the number of the p-values that are smaller than the probability of Type I error α.

Table 8 shows results for tests at $\alpha = 0.01$, 0.05 and 0.10. We find that, for all models, the Type I error rates obtained from the MCMC p-values are very close to α even when they are calculated from 1,000 random tables, while the rates from the asymptotic p-values greatly differ from each α. As expected, the asymptotic χ^2 test is not reliable for such sparse tables.

Figure 3 shows the plots of the MCMC and asymptotic p-values for each of the decomposable log-linear models. We can see that the MCMC p-values are almost identical to α, because they are arranged very close to or on the bisecting line. These results show that the proposed M–H algorithm for testing decomposable log-linear

Table 5 Presence or absebce of turus mandibularis in three Aleutian populations from Muller and Mayhall (1971)

Population (X_1)	Sex (X_2)	Presence or absence (X_3)	Age group (X_4) 1–10	11–20	21–30	31–40	41–50	50+
Lgloolik	Male	Present	4	8	13	18	10	12
		Absent	44	32	21	5	0	1
	Female	Present	1	11	19	13	6	10
		Absent	42	17	17	5	4	2
Hall Beach	Male	Present	2	5	7	5	4	4
		Absent	17	10	6	2	2	1
	Female	Present	1	3	2	5	4	2
		Absent	12	16	6	2	0	0
Aleut	Male	Present	4	2	4	7	4	3
		Absent	6	13	3	3	5	3
	Female	Present	3	1	2	2	2	4
		Absent	10	7	12	5	2	1

Table 6 Asymptotic p-values versus MCMC p-values

Clique set	χ^2 value (df)	Asymptotic p-value	MCMC p-value Proposed M–H	Dobra's M–H
$\mathcal{C}_1 = \{\{1, 4\}, \{2, 4\}, \{3, 4\}\}$	61.315 (42)	0.02737	0.03056 (0.00117)	0.02952 (0.00185)
$\mathcal{C}_2 = \{\{1, 4\}, \{2, 3\}, \{3, 4\}\}$	64.073 (46)	0.04009	0.03989 (0.00093)	0.03909 (0.00225)
$\mathcal{C}_3 = \{\{1, 2\}, \{2, 4\}, \{3, 4\}\}$	72.183 (50)	0.02169	0.01929 (0.00129)	0.02522 (0.00176)
$\mathcal{C}_4 = \{\{1, 3\}, \{2, 4\}, \{3, 4\}\}$	70.748 (50)	0.02828	0.02356 (0.00199)	0.03278 (0.00195)
$\mathcal{C}_5 = \{\{1, 3\}, \{2, 3\}, \{3, 4\}\}$	73.659 (54)	0.03890	0.04067 (0.00196)	0.04026 (0.00244)
$\mathcal{C}_6 = \{\{1, 2\}, \{2, 3\}, \{3, 4\}\}$	75.383 (54)	0.02886	0.03025 (0.00134)	0.02838 (0.00189)

The values in parenthesis are the standard errors of the MCMC p-values

models is a valuable alternative to the asymptotic chi-squared test for sparse contingency tables.

7 Concluding remarks

In this study, we showed that a Markov basis on the junction tree for a decomposable log-linear model can be expressed as the union of Markov bases characterized by

Table 7 Probabilities of decomposable log-linear models

$\mathcal{C}_1 = \{\{1,3\},\{2,3\},\{3,4\}\}$

			X_4	
X_1	X_2	X_3	0	1
1	1	1	0.050	0.044
		2	0.075	0.050
	2	1	0.040	0.035
		2	0.143	0.097
2	1	1	0.082	0.072
		2	0.039	0.026
	2	1	0.065	0.057
		2	0.075	0.050

$\mathcal{C}_2 = \{\{1,4\},\{2,4\},\{3,4\}\}$

			X_4	
X_1	X_2	X_3	0	1
1	1	1	0.049	0.056
		2	0.068	0.060
	2	1	0.061	0.074
		2	0.085	0.080
2	1	1	0.057	0.033
		2	0.080	0.036
	2	1	0.071	0.044
		2	0.099	0.047

$\mathcal{C}_3 = \{\{1,4\},\{2,4\},\{2,3\}\}$

			X_4	
X_1	X_2	X_3	0	1
1	1	1	0.066	0.066
		2	0.051	0.050
	2	1	0.051	0.054
		2	0.094	0.100
2	1	1	0.078	0.039
		2	0.059	0.030
	2	1	0.059	0.032
		2	0.110	0.059

$\mathcal{C}_4 = \{\{1,3\},\{3,4\},\{2,4\}\}$

			X_4	
X_1	X_2	X_3	0	1
1	1	1	0.040	0.034
		2	0.097	0.063
	2	1	0.050	0.045
		2	0.120	0.084
2	1	1	0.066	0.055
		2	0.051	0.033
	2	1	0.082	0.073
		2	0.063	0.044

Table 8 Type I error rates at $\alpha = 0.01$, 0.05 and 0.10

Clique set	$\alpha = 0.01$		$\alpha = 0.05$		$\alpha = 0.10$	
	Asymptotic	MCMC	Asymptotic	MCMC	Asymptotic	MCMC
$\mathcal{C}_1$	0.023	0.015	0.095	0.054	0.172	0.105
$\mathcal{C}_2$	0.020	0.009	0.082	0.044	0.153	0.092
$\mathcal{C}_3$	0.025	0.012	0.094	0.056	0.167	0.102
$\mathcal{C}_4$	0.023	0.013	0.104	0.057	0.194	0.113

global Markov properties. We provided an explicit form of the Markov basis given in Definition 3 and then described **Random table generation** for generating Markov bases. We also presented the M–H algorithm with a Markov basis for generating contingency tables of a Markov chain. The M–H algorithm using **Random table generation** allows the production of contingency tables globally in the set of contingency tables with fixed marginals.

In the numerical experiments, we demonstrated that, for decomposable log-linear models for sparse contingency tables, the M–H algorithm using **Random table generation** enables finding more reliable p-values than the asymptotic chi-squared approximation and generates random tables more efficiently than Dobra's M–H algorithm in terms of the inefficiency factor.

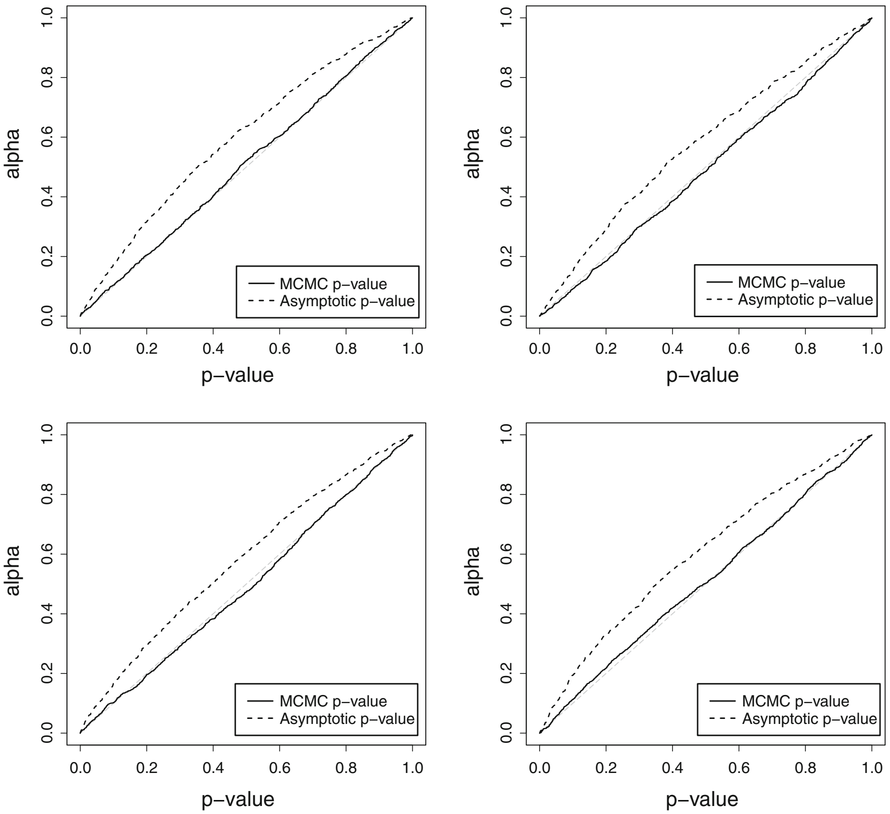

Fig. 3 Plots of the MCMC and asymptotic p-values for the decomposable log-linear models with junction trees specified by $\mathcal{C}_1$ to $\mathcal{C}_4$ given in Table 7 (starting from *top left* to *right*)

Moreover, we found that the proposed M–H algorithm works well to test decomposable log-linear models for sparse contingency tables.

Acknowledgments The authors would like to thank the editor and referees whose valuable comments and kind suggestions that led to an improvement of this paper. This research is supported by the Japan Society for the Promotion of Science (JSPS), Grant-in-Aid for Scientific Research (C), No. 20500263.

Appendix: Configuration A of the conditional independence model for a three-way contingency table

We consider a $2 \times 2 \times 3$ table cross-classified by $X = (X_1, X_2, X_3)$. For the table, we assume the conditional independence that $X_1 \perp X_2 | X_3$. When we consider the reverse lexicographic order of the cell indices in $\{1, 2\} \times \{1, 2\} \times \{1, 2, 3\}$, that is, $(1, 1, 1), (2, 1, 1), \ldots, (2, 2, 3)$, A for the conditional independence model is given by

$$A = \begin{pmatrix} E_3 \otimes \mathbf{1}_2^\top \otimes E_1 \\ E_3 \otimes E_2 \otimes \mathbf{1}_1^\top \end{pmatrix} = \begin{pmatrix} 1&0&1&0&0&0&0&0&0&0&0&0 \\ 0&1&0&1&0&0&0&0&0&0&0&0 \\ 0&0&0&0&1&0&1&0&0&0&0&0 \\ 0&0&0&0&0&1&0&1&0&0&0&0 \\ 0&0&0&0&0&0&0&0&1&0&1&0 \\ 0&0&0&0&0&0&0&0&0&1&0&1 \\ 1&1&0&0&0&0&0&0&0&0&0&0 \\ 0&0&1&1&0&0&0&0&0&0&0&0 \\ 0&0&0&0&1&1&0&0&0&0&0&0 \\ 0&0&0&0&0&0&1&1&0&0&0&0 \\ 0&0&0&0&0&0&0&0&1&1&0&0 \\ 0&0&0&0&0&0&0&0&0&0&1&1 \end{pmatrix}.$$

By permuting rows of A, we have the block diagonal matrix

$$A' = \left(\begin{array}{cccccccccccc} 1&0&1&0&0&0&0&0&0&0&0&0 \\ 0&1&0&1&0&0&0&0&0&0&0&0 \\ 1&1&0&0&0&0&0&0&0&0&0&0 \\ 0&0&1&1&0&0&0&0&0&0&0&0 \\ \hline 0&0&0&0&1&0&1&0&0&0&0&0 \\ 0&0&0&0&0&1&0&1&0&0&0&0 \\ 0&0&0&0&1&1&0&0&0&0&0&0 \\ 0&0&0&0&0&0&1&1&0&0&0&0 \\ \hline 0&0&0&0&0&0&0&0&1&0&1&0 \\ 0&0&0&0&0&0&0&0&0&1&0&1 \\ 0&0&0&0&0&0&0&0&1&1&0&0 \\ 0&0&0&0&0&0&0&0&0&0&1&1 \end{array}\right) = \begin{pmatrix} A_{22} & & 0 \\ & A_{22} & \\ 0 & & A_{22} \end{pmatrix} = \begin{pmatrix} A'_1 \\ A'_2 \\ A'_3 \end{pmatrix}.$$

References

Blair JRS, Barry P (1993) An introduction to chordal graphs and clique trees. In: Graph theory and sparse matrix computation. IMA vol Math Appl 56. Springer, New York, pp 1–30

Dalenius P, Reiss RS (1982) Data-swapping: a technique for disclosure control. J Stat Plan Inference 6:73–85

Diaconis P, Sturmfels B (1998) Algebraic algorithms for sampling from conditional distributions. Ann Stat 26:363–397

Dobra A (2003) Markov bases for decomposable graphical models. Bernoulli 9:1093–1108

Geiger D, Meek Ch, Sturmfels B (2006) On the toric algebra of graphical models. Ann Stat 34:1463–1492

Geyer CJ (1992) Practical Markov Chain Monte Carlo. Stat Sci 7:473–483

Goorin AM, Perez-Atayde A, Gebhardt M, Andersen JW, Wilkinson RH, Delorey MJ, Watts H, Link M, Jaffe N, Frei DE III (1987) Weekly high-dose methotrexate and doxorubicin for osteosarcoma: the Dana-Farber Cancer Institute/the Children's Hospital-study III. J Clin Oncol 5:1178–1184

Hájek P, Havránek T, Jiroušek R (1992) Uncertain information processing in expert systems. CRC Press, Boca Raton

Hara H, Aoki S, Takemura A (2010) Minimal and minimal invariant Markov bases of decomposable models for contingency tables. Bernoulli 16:208–233

Hastings WK (1970) Monte Carlo sampling methods using Markov chains and their applications. Biometrika 57:97–109

Innes JR, Ulland BM, Valerio MG, Petrucelli L, Fishbein L, Hart ER, Pallotta AJ, Bates RR, Falk HL, Gart JJ, Klein M, Mitchell I, Peters J (1969) Bioassay of pesticides and industrial chemicals for tumorigenicity in mice: a preliminary note. J Natl Cancer Inst 42:1101–1114

Jensen FV, Jensen F (1994) Optimal junction trees. In: de Mantaras RL, Poole D (eds) Proceedings of the 10th conference on uncertainty in artificial intelligence. Morgan Kaufmann, San Mateo, pp 360–366
Jones GL, Haran M, Caffo BS, Neath R (2006) Fixed width output analysis for Markov chain Monte Carlo. J Am Stat Assoc 101:1537–1547
Lauritzen SL (1996) Graphical models. Oxford University Press, New York
Muller T, Mayhall JT (1971) Analysis of contingency table data on torus mandibularis using a log linear model. Am J Phys Anthropol 34:149–153