Performing the Exact Test of Hardy-Weinberg Proportion for Multiple Alleles

Sun Wei Guo; Elizabeth A. Thompson

Biometrics, Vol. 48, No. 2. (Jun., 1992), pp. 361-372.

Stable URL:
http://links.jstor.org/sici?sici=0006-341X%28199206%2948%3A2%3C361%3APTETOH%3E2.0.CO%3B2-D

Biometrics is currently published by International Biometric Society.

BIOMETRICS 48, 361–372
June 1992

Performing the Exact Test of Hardy–Weinberg Proportion for Multiple Alleles

Sun Wei Guo[1,*] and Elizabeth A. Thompson[1,2]

[1] Department of Biostatistics, SC-32,
[2] Department of Statistics, GN-22,
University of Washington, Seattle, Washington 98195, U.S.A.

SUMMARY

The Hardy–Weinberg law plays an important role in the field of population genetics and often serves as a basis for genetic inference. Because of its importance, much attention has been devoted to tests of Hardy–Weinberg proportions (HWP) over the decades. It has long been recognized that large-sample goodness-of-fit tests can sometimes lead to spurious results when the sample size and/or some genotypic frequencies are small. Although a complete enumeration algorithm for the exact test has been proposed, it is not of practical use for loci with more than a few alleles due to the amount of computation required. We propose two algorithms to estimate the significance level for a test of HWP. The algorithms are easily applicable to loci with multiple alleles. Both are remarkably simple and computationally fast. Relative efficiency and merits of the two algorithms are compared. Guidelines regarding their usage are given. Numerical examples are given to illustrate the practicality of the algorithms.

1. Introduction

The importance of the Hardy–Weinberg law in the development of population genetics cannot be overstated (see, for example, Crow, 1988). This law says that in a large random-mating population with no selection, mutation, or migration, the allele frequencies and the genotype frequencies are constant from generation to generation and that, furthermore, there is a simple relationship between the allele frequencies and the genotype frequencies. For an m-allele autosomal locus with alleles $A_1, A_2, \ldots, A_m$, the genotypic array as given by the Hardy–Weinberg law is

$$\sum_i p_i^2 A_i A_i + \sum_{i<j} 2p_i p_j A_i A_j,$$

where p_i is the allelic frequency of A_i. A population with these genotype frequencies is said to be in Hardy–Weinberg equilibrium at the locus under investigation; the genotype frequencies are known as Hardy–Weinberg proportions (HWP). Because of its importance, testing of the hypothesis that a population exhibits HWP has drawn a lot of attention during past decades, though the problem itself seems to be very simple from a statistical viewpoint.

The methods proposed so far for testing HWP can be categorized into two groups. One consists of large-sample goodness-of-fit tests such as Pearson's χ^2, likelihood ratio statistic

* *Current address:* Department of Biostatistics, University of Michigan, 109 S. Observatory, Ann Arbor, Michigan 48109, U.S.A.

Key words: Exact tests; Hardy–Weinberg; Markov chain; Metropolis algorithm; Monte Carlo; Multiple alleles.

G^2, and conditional χ^2 test (Li, 1955). These tests have one characteristic in common: They lean heavily on asymptotic results. The other approach involves exact tests (Levene, 1949; Haldane, 1954; Chapco, 1976). These tests usually involve a lot of computing and were, in the past, thus restricted to diallelic locus with small sample sizes due to lack of computing power. It has long been recognized that the standard goodness-of-fit tests can sometimes lead to false rejection or acceptance of HWP when the sample sizes are small and/or some cell frequencies are small or zero (see, for example, Emigh, 1980). Although various corrections for small sample sizes are made (Emigh and Kempthorne, 1975; Elston and Forthofer, 1977; Smith, 1986), it is found that they usually do not greatly improve the results obtained from the traditional goodness-of-fit tests (Emigh, 1980; Hernández and Weir, 1989). Thus an exact test is preferred when the sample size is small and/or some cell frequencies are small or zero.

On the other hand, with the advent of the restriction fragment length polymorphism (RFLP) and variable number of tandem repeats (VNTRs), genetic loci with 10 or more alleles are not uncommon nowadays (Botstein et al., 1980; Nakamura et al., 1987). Hence, even if the sample size is moderately large, the number of genotypes is so large that some sample genotype frequencies will be zero, especially when the corresponding population allele frequencies are low (Figure 1). As a result, the adequacy of applying classical goodness-of-fit tests of HWP is questionable and use of the exact test is desirable.

A_1	3							
A_2	4	2						
A_3	2	2	2					
A_4	3	3	2	1				
A_5	0	1	0	0	0			
A_6	0	0	0	0	0	1		
A_7	0	0	1	0	0	0	0	
A_8	0	0	0	2	1	0	0	0
	A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8

Figure 1. A sample of size 30 of genotype frequencies simulated under HWP when the underlying gene frequencies are (.2, .2, .2, .2, .05, .05, .05, .05).

Louis and Dempster (1987) proposed an algorithm for generating the exact distribution of a finite sample drawn from a population in HWP. The algorithm works well when the number of alleles is low (say, four or five) but it will be too computer-intensive to be of practical use when there are many alleles since the number of possible samples with same gene frequencies and sample sizes grows exponentially with the number of alleles (Hernández and Weir, 1989). To avoid complete enumeration, Hernández and Weir suggest use of a conventional Monte Carlo method to obtain an estimated *P*-value. So far, however, a general algorithm to perform the test of HWP using Monte Carlo methods has not been available.

In this paper, we present two methods to estimate the exact significance levels of the exact test for HWP for multiple alleles. One is a conventional Monte Carlo method, due to Joe Felsenstein (personal communication), and the other is an adaptation of the Metropolis algorithm long known in statistical physics (Metropolis et al., 1953; Binder and Heermann, 1988). Whereas the former method takes advantage of the fact that random mating can be regarded as random union of two gametes, the crux of the latter is the construction of a Markov chain with equilibrium distribution matching the genotype probabilities under HWP of samples that have the same allelic counts as the observed data. The method can be regarded as a variant of the one proposed by Guo and Thompson (Technical Report #187, Department of Statistics, University of Washington, 1989) for the analysis of sparse contingency tables, which is, in turn, an extension of the approach of Besag and Clifford (1989) for binary tables. In Section 2 we give some notation and Levene's (1949) distribution. The algorithms are presented in Sections 3 and 4, respectively. The practicality of the method is demonstrated by several examples in Section 5. The relative merits of two methods are discussed in Section 6. The programs implementing the methods proposed in this paper are written in C, to run under BSD4.3 UNIX. These programs are available free of charge from the authors upon request.

2. Exact Test for *m* Alleles

Consider an autosomal locus that has m alleles $A_1, A_2, \ldots, A_m$. If a sample of size n is sampled from a population of interest, the data can be presented as the array

A_1	f_{11}			
A_2	f_{21}	f_{22}		
$\vdots$	$\cdots$	$\cdots$	$\cdots$	
A_m	f_{m1}	f_{m2}	$\cdots$	f_{mm}
	A_1	A_2	$\cdots$	A_m

where f_{ij} $(1 \leqslant j \leqslant i \leqslant m)$ is the observed count of genotype A_iA_j. Throughout the paper, we will use $\mathbf{f} = (f_{11}, f_{21}, f_{22}, \ldots, f_{mm})$ to designate this table. If $f_i = f_{ii} + f_{i+} = f_{ii} + \sum_{j=1}^{k} f_{ij}$ (where $f_{ij} = f_{ji}$ if $j > i$), then f_i is the number of A_i alleles in the sample. Then, under HWP and conditional on $\{f_i\}$, the probability of obtaining the sample $\mathbf{f}$ is (Levene, 1949):

$$\Pr(\mathbf{f}) = \frac{n! \prod_{i=1}^{m} f_i!}{(2n)! \prod_{j>i} f_{ij}!} 2^{\sum_{j>i} f_{ij}}. \tag{1}$$

The exact test for HWP given observed sample $\mathbf{f}$ has to evaluate

$$P = \sum_{\mathbf{g} \in \mathscr{S}} \Pr(\mathbf{g}), \tag{2}$$

where $\mathscr{S} = \{\mathbf{g}: \Pr(\mathbf{g}) \leqslant \Pr(\mathbf{f}), \mathbf{g} \in \Gamma_0\}$, and

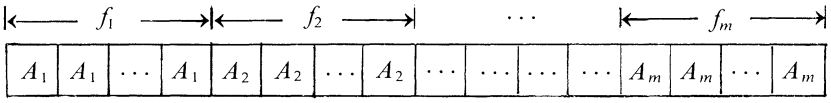
$$\Gamma_0 = \Gamma(\mathbf{f}) = \{\mathbf{g}: \mathbf{g} \text{ has the same allele counts } \{g_i\} \text{ as does } \mathbf{f}\}.$$

Rejection or acceptance of the null hypothesis depends on whether P is smaller than a prespecified significance level α.

3. Conventional Monte Carlo Method

Suppose a sample of size n is drawn from a population of interest and we have data $\mathbf{f}$. We can visualize the genotypes of these n individuals formed by f_i A_i-gametes $(i = 1, \ldots, m)$

in a particular manner. Under HWP, a sample of size n with f_i A_i-gametes ($i = 1, \ldots, m$) can be regarded as n random unions of two gametes. Thus, if the $2n$ gametes are first arranged in the following manner,



and then a random permutation is applied, the n successive pairs of alleles can be regarded as a sample of size n drawn from a population under HWP, with f_i gametes of type A_i ($i = 1, \ldots, m$). If we permute and chop it into n pairs again, then the n pairs can be regarded as another random sample from the same population. Thus we have the following algorithm to estimate the P-value:

1. Compute Pr(**f**); set counter $K = 0$.
2. Construct index vector s, with $s_1 = s_2 = \cdots = s_{f_1} = 1$, $s_{f_1+1} = \cdots = s_{f_1+f_2} = 2$, $\ldots$, $s_{f_1+f_2+\cdots+f_{m-1}+1} = \cdots = s_{f_m} = m$.
3. For specified simulation size N do the following:
   (i) Reset s to a random permutation of previous s.
   (ii) Obtain sample **g** by chopping s into n consecutive pairs and letting $\mathbf{g} = \{A_{s_{2k-1}}A_{s_{2k}}, k = 1, 2, \ldots, n\}$.
   (iii) Calculate Pr(**g**).
   (iv) If Pr(**g**) $\leq$ Pr(**f**), add 1 to K.
4. The Monte Carlo P-value is K/N.

Since each drawing is independent, we can calculate approximate simulation size N to ensure that the estimated P-value is within δ units of the true one with $(1 - \gamma)100\%$ confidence (Agresti, Wackerly, and Boyett, 1979). In fact, for any P-value, if $Z_{\gamma/2}$ denotes the $(1 - \gamma/2)$th quantile of the standard normal distribution,

$$N \geq \left(\frac{Z_{\gamma/2}}{2\delta}\right)^2$$

will be sufficient. For example, if one wants to estimate P-value to within $\delta = .01$ with 99% confidence,

$$N \geq \left(\frac{2.576}{(2)(.01)}\right)^2 = 16{,}589.44 \approx 17{,}000$$

4. Markov Chain Method

4.1 *Construction of Markov Chain*

In order to define a Markov chain on the space Γ_0, corresponding to the observed allelic counts, we need to specify the neighboring states for any $\mathbf{f} \in \Gamma_0$, i.e., other states to which the system can jump by one-step transitions. To do this, consider any $\mathbf{f} \in \Gamma_0$. For notational convenience, we will, throughout the paper, write f_{ij} even if $j > i$ but we will view f_{ij} as f_{ji} if $j > i$.

Let (i_1, i_2) and (j_1, j_2) be two integer pairs, with $1 \leq i_1 < i_2 \leq m$ and $1 \leq j_1 < j_2 \leq m$. Define $\Delta = \delta_{i_1j_1} + \delta_{i_1j_2} + \delta_{i_2j_1} + \delta_{i_2j_2}$ and $\delta = \delta_{i_1j_1}\delta_{i_2j_2}$, where δ_{ij} is the usual Kronecker function. Obviously, $0 \leq \Delta \leq 2$. For two given integer pairs, there are three mutually

exclusive cases:

(i) $\Delta = 0$

If $f_{i_1j_1} f_{i_2j_2} \neq 0$, we can subtract 1 from $f_{i_1j_1}$ and from $f_{i_2j_2}$, and add 1 to $f_{i_1j_2}$ and to $f_{i_2j_1}$, respectively. We refer to this process as type 0 donation switch or, in short, D_0-switch (with respect to $f_{i_1j_1}$) and say **f** is D_0-switchable. Similarly, we can define a type 0 reception switch or R_0-switch if $f_{i_1j_2} f_{i_2j_1} \neq 0$; in this case, $f_{i_1j_1}$ and $f_{i_2j_2}$ are increased by 1, and $f_{i_1j_2}$ and $f_{i_2j_1}$ are each decreased by 1.

(ii) $\Delta = 1$

If $f_{i_1j_1} f_{i_2j_2} \neq 0$, we can subtract 1 from $f_{i_1j_1}$ and from $f_{i_2j_2}$, and add 1 to $f_{i_1j_2}$ and to $f_{i_2j_1}$, respectively. We refer to this process as type 1 donation switch or, in short, D_1-switch (with respect to $f_{i_1j_1}$) and say **f** is D_1-switchable. Similarly, we can define a type 1 reception switch or R_1-switch if $f_{i_1j_2} f_{i_2j_1} \neq 0$. (Algebraically, this case is identical to the case $\Delta = 0$, but geometrically in the triangular array, it differs: One homozygous genotype is involved in the switch.)

(iii) $\Delta = 2$

In this case $i_1 = j_1$ and $i_2 = j_2$: Two homozygous genotypes are involved. If $f_{i_1j_1} f_{i_2j_2} \neq 0$ we can subtract 1 from $f_{i_1j_1}$ and from $f_{i_2j_2}$, and add 2 to $f_{i_1j_2}$. This process is called a type 2 donation switch or D_2-switch (with respect to $f_{i_1j_1}$). In similar fashion, a type 2 reception switch or R_2-switch can be defined if $f_{i_1j_2} \geq 2$.

Regardless of type, if, for given integer pairs (i_1, i_2) and (j_1, j_2), **f** is either D-switchable or R-switchable, but not both, **f** is said to be partially switchable (PS). If **f** is both D-switchable and R-switchable, we say **f** is fully switchable (FS). If **f** is neither PS nor FS, **f** is said to be nonswitchable (NS).

Note that after a switch is made (assuming switchable), we obtain a new table, say **g**, which has the same allele counts as that of **f**. That is, $\mathbf{g} \in \Gamma(\mathbf{f})$. Further, for any $\mathbf{g} \in \Gamma(\mathbf{f})$, if a switch is made on **g** so that a new table $\mathbf{g}'$ is obtained, then $\mathbf{g}' \in \Gamma(\mathbf{f})$ and the probability ratio $P(\mathbf{g}')/P(\mathbf{g})$ depends only on those cell entries of **g** that are changed. Table 1 shows these ratios for different switches.

By the Metropolis algorithm (Metropolis et al., 1953), we can construct a Markov chain $\{\mathbf{f}(t); t = 1, 2, \ldots\}$ of genotype counts that have the same allelic counts as the observed sample, **f**, and an equilibrium distribution $\Pr(\mathbf{f})$, the probablity of **f** under HWP. The state space for the Markov chain is $\Gamma_0 = \Gamma(\mathbf{f})$, a finite subset of the set of all triangular arrays of counts $\{\mathbf{f}^{(k)}: k = 1, 2, \ldots\}$. Suppose a state $\mathbf{f}^{(k)} \in \Gamma_0$ is given. A new state $\mathbf{f}^{(l)}$ is proposed

Table 1
Probability ratios for different switches ($\gamma = 2^{\delta}(\frac{1}{2})^{1-\delta}$)

Type of switch	Probability ratio	Requirement
D_0	$f_{i_1j_1} f_{i_2j_2} / [(f_{i_1j_2} + 1)(f_{i_2j_1} + 1)]$	$f_{i_1j_1} f_{i_2j_2} \neq 0$
R_0	$f_{i_1j_2} f_{i_2j_1} / [(f_{i_1j_1} + 1)(f_{i_2j_2} + 1)]$	$f_{i_1j_2} f_{i_2j_1} \neq 0$
D_1	$\gamma f_{i_1j_1} f_{i_2j_2} / [(f_{i_1j_2} + 1)(f_{i_2j_1} + 1)]$	$f_{i_1j_1} f_{i_2j_2} \neq 0$
R_1	$f_{i_1j_2} f_{i_2j_1} / [(\gamma(f_{i_1j_1} + 1)(f_{i_2j_2} + 1)]$	$f_{i_1j_2} f_{i_2j_1} \neq 0$
D_2	$4f_{i_1j_1} f_{i_2j_2} / [(f_{i_1j_2} + 2)(f_{i_1j_2} + 1)]$	$f_{i_1j_1} f_{i_2j_2} \neq 0$
R_2	$f_{i_1j_2}(f_{i_1j_2} - 1) / [4(f_{i_1j_1} + 1)(f_{i_2j_2} + 1)]$	$f_{i_1j_2} \geq 2$

with probability Q_{kl}. If $\mathbf{f}^{(l)} \notin \Gamma_0$, $Q_{kl} = 0$. The Markov chain transition probabilities are (Ripley, 1987):

$$\Pr(\mathbf{f}^{(k)} \to \mathbf{f}^{(l)}) = \min\left(1, \frac{\Pr(\mathbf{f}^{(l)})}{\Pr(\mathbf{f}^{(k)})}\right) Q_{kl} \quad (k \neq l), \tag{3}$$

$$\Pr(\mathbf{f}^{(k)} \text{ unchanged}) = 1 - \sum_{l \neq k} \min\left(1, \frac{\Pr(\mathbf{f}^{(l)})}{\Pr(\mathbf{f}^{(k)})}\right) Q_{kl}. \tag{4}$$

For any two given integer pairs (i_1, i_2) and (j_1, j_2) with $1 \leq i_1 < i_2 \leq m$ and $1 \leq j_1 < j_2 \leq m$, there are three situations that we have discussed before: $\mathbf{f}^{(k)}$ is NS, PS, or FS. If $\mathbf{f}^{(k)}$ is NS, then we choose $Q_{kl} = 0$ for $l \neq k$; if $\mathbf{f}^{(k)}$ is PS and can be switched to, say, $\mathbf{f}^{(d)}$, then we choose $Q_{kl} = 0$ for $l \neq k, d$ and let $Q_{kd} = Q_{kk} = \frac{1}{2}$; if $\mathbf{f}^{(k)}$ is FS and can be switched to, say $\mathbf{f}^{(d)}$ and $\mathbf{f}^{(e)}$, then we let $Q_{kd} = Q_{ke} = \frac{1}{2}$ and $Q_{kj} = 0$ for $j \neq d, e$.

The Markov chain we thus construct is finite and irreducible. The irreducibility comes from the fact that for any two tables in $\Gamma(\mathbf{f})$, one can be obtained from another through a sequence of switches. (For proof, see Appendix.)

4.2 *The Basic Formulation of the Method*

It is instructive to rewrite (2) as

$$P = \sum_{\mathbf{g} \in \mathscr{S}} \Pr(\mathbf{g}) = \sum_{\mathbf{g} \in \Gamma_0} I_{[\Pr(\mathbf{g}) \leq \Pr(\mathbf{f})]} \Pr(\mathbf{g}) = \mathrm{E}(h(\mathbf{g})), \tag{5}$$

where I is an indicator function, $h(\mathbf{g}) = I_{[\Pr(\mathbf{g}) \leq \Pr(\mathbf{f})]}$, and $\mathbf{f}$ is the observed table.

Equation (5) says that P is exactly the expectation of the indicator function on space Γ_0. Hence, for any function $h(\mathbf{f})$ defined on Γ_0, if we simulate this Markov chain for $t = 1, 2, \ldots, N$, the average

$$\hat{F} = \sum_{t=1}^{N} h(\mathbf{f}(t))/N \tag{6}$$

is an estimate of $\mathrm{E}(h(\mathbf{f}))$.

For finite irreducible Markov chains, $\hat{F}$ is asymptotically normally distributed and $\hat{F} \to \mathrm{E}(h(\mathbf{f}))$ with probability 1 as $N \to \infty$ (Hastings, 1970).

In order for the estimate to be unbiased, it is ideal to start the chain from a random state chosen from the distribution $\Pr(\mathbf{f})$. If the chain is started from an arbitrary state (e.g., the observed state), then the estimate will be biased, especially if the starting state is in a region of low probability. One remedy is to start the chain from its initial state and run for a long time so that the initial state is "forgotten." We refer to this process as "dememorization."

Based on these facts, we have the following algorithm to estimate P:

1. Let $\mathbf{g} = \mathbf{f}$; set counter K to 0 and cumulative ratio, ρ, to 1.

2. *Dememorization:* For prespecified number of steps M, do the following:

(i) *Pick:* Randomly pick two integer pairs, (i_1, i_2) and (j_1, j_2), with $1 \leq i_1 < i_2 \leq m$ and $1 \leq j_1 < j_2 \leq m$.

(ii) *Switch:* Determine the switchability of $\mathbf{g}$ and switch with transition probability calculated from (3):

—If $\mathbf{g}$ is NS, the table remains unchanged and $\rho' = \rho$.

—If $\mathbf{g}$ is PS, then switch $\mathbf{g} \to \mathbf{g}'$, say, with probability $\Pr(\mathbf{g} \to \mathbf{g}')$ and remain the same pattern unchanged with probability $1 - \Pr(\mathbf{g} \to \mathbf{g}')$. Let $\rho' = c\rho$, where $c = \Pr(\mathbf{g}')/\Pr(\mathbf{g})$, if $\mathbf{g}$ is switched to $\mathbf{g}'$, or $\rho' = \rho$ if $\mathbf{g}$ is unchanged.

—If $\mathbf{g}$ is FS, and $\mathbf{g}$ can be switched to $\mathbf{g}'$ and $\mathbf{g}''$, say, then $\mathbf{g} \to \mathbf{g}'$ with probability $\Pr(\mathbf{g} \to \mathbf{g}')$, $\mathbf{g} \to \mathbf{g}''$ with probability $\Pr(\mathbf{g} \to \mathbf{g}'')$, or remains unchanged with probability $1 - \Pr(\mathbf{g} \to \mathbf{g}') - \Pr(\mathbf{g} \to \mathbf{g}'')$. Let $\rho' = c\rho$, where $c = \Pr(\mathbf{g}')/\Pr(\mathbf{g})$, $\Pr(\mathbf{g}'')/\Pr(\mathbf{g})$, or 1, depending on whether $\mathbf{g}$ is switched to $\mathbf{g}'$, $\mathbf{g}''$, or remained unchanged.

(iii) *Update:* Let $\rho = \rho'$, and the new table be $\mathbf{g}$ if a switch is made.

3. *Estimation:* For prespecified number of steps N do the following:

(i) *Pick and switch*, as in step 2.
(ii) *Count:* If $\rho' \leqslant 1$, add 1 to K.
(iii) *Update:* Let $\rho = \rho'$, and the new table be $\mathbf{g}$ if a switch is made.

4. The Monte Carlo test P-value is K/N.

To prevent numerical overflow and underflow in computation, we suggest using $\log c = \log[\Pr(\mathbf{g}')/\Pr(\mathbf{g})]$, and $\log \rho' = \log \rho + \log c$.

Since the two neighboring states are correlated, the usual estimation of statistical variation does not apply to this case. However, we can use a batching method to get an approximate estimate (Hastings, 1970; Ripley, 1987): Divide the observation into B batches of C consecutive observations each, and compute the mean of the ith batch by

$$\bar{h}_i = \sum_{t=1}^{C} h((i-1)C + t)/C$$

and use

$$S^2 = \sum_{i=1}^{B} (\bar{h}_i - \bar{h})^2/[B(B-1)]$$

as an estimate of variance.

5. Numerical Examples

In this section we provide some examples using the methods proposed in Sections 3 and 4. The goodness-of-fit tests were done using functions written in S (Becker, 1988). All the computations were done on an IBM/RT workstation which uses BSD4.3 UNIX. The random number generator used was the run-time library *rans*, which produces pseudo-random numbers in the interval (0, 1).

Example 1. We reanalyze the data in Figure 2 for which the exact P-value was reported by Louis and Dempster (1987). It took, as they reported, 101.34 CPU seconds to come up

	A_1	A_2	A_3	A_4
A_1	0			
A_2	3	1		
A_3	5	18	1	
A_4	3	7	5	2

Figure 2. Genotype frequency data from Louis and Dempster (1987).

with the exact *P*-value of .01744 on an IBM 3081 computer using a complete enumeration algorithm proposed by Louis and Dempster. The results of various goodness-of-fit tests, along with the results of exact tests, are given in Table 2. Sample sizes of 1,700 and 17,000 were used for the conventional Monte Carlo method to ensure that the estimate is within .02 with 90% confidence and within .01 with 99% confidence, respectively.

Note that none of the *P*-values calculated from goodness-of-fit tests is very close to the true one, with the *P*-value given by Pearson's statistic with continuity correction .5 being 10 times as large as that calculated via the likelihood ratio statistic. Note also that in this example the Pearson statistic with a continuity correction of .5 yields the most conservative result.

Example 2. We analyze the simulated data that are presented in Section 1. The results of goodness-of-fit and exact tests are listed in Table 3. It is interesting to note that the different statistics employed could lead to completely different conclusions. The *P*-values calculated from Pearson's statistics, with and without continuity corrections, are erroneously small, with $\chi^2_{.5}$ being the most extreme. In retrospect, this is not surprising. For multiple alleles, when all the allele frequencies are low and some extremely low ($\leq .05$, say), the expected values can be much smaller than .5 or .25, resulting in erratically inflated values of the χ^2, $\chi^2_{.5}$, or $\chi^2_{.25}$ statistics.

Table 2

Results of goodness-of-fit and exact tests. F^2 is the Freeman–Tukey statistic; G^2, the likelihood ratio statistic; χ^2, Pearson's statistic; $\chi^2_{.5}$, Pearson's statistic with continuity correction of .5; $\chi^2_{.25}$, *Pearson's statistic with continuity correction of* .25 *as suggested by Emigh* (1980). *B is the number of batches, C is the size of each batch. For the Markov chain method, the dememorization period is* 1,000 *steps. The CPU time is in seconds (on an IBM/RT, unless otherwise specified).*

Statistic/Method	Value	*P*-value	S.E.	CPU time
F^2	16.2761	.01235	—	—
G^2	17.1828	.008634	—	—
χ^2	14.6270	.02337	—	—
$\chi^2_{.5}$	11.0156	.08789	—	—
$\chi^2_{.25}$	12.5852	.05012	—	—
Complete enumeration	—	.01744	—	101.34 (on IBM 3081)
Monte Carlo ($B = 10$, $C = 170$)	—	.01706	.003557	19.4
Monte Carlo ($B = 100$, $C = 170$)	—	.01724	.000954	193.5
Markov chain ($B = 20$, $C = 1{,}000$)	—	.01705	.003617	17.2
Markov chain ($B = 20$, $C = 10{,}000$)	—	.01727	.001404	165.2

Table 3

Results of goodness-of-fit and exact tests for simulated 8*-allele data. The dememorization period is* 1,000 *steps long for Markov chain method.*

Statistic/Method	Value	*P*-value	S.E.	CPU time
F^2	14.7601	.9809	—	—
G^2	25.9748	.5744	—	—
χ^2	51.9302	.003908	—	—
$\chi^2_{.5}$	71.5300	1.1325×10^{-5}	—	—
$\chi^2_{.25}$	39.4041	.07464	—	—
Monte Carlo ($B = 10$, $C = 170$)	—	.2182	.008058	14.3
Monte Carlo ($B = 100$, $C = 170$)	—	.2181	.003242	142.9
Markov chain ($B = 40$, $C = 5{,}000$)	—	.2186	.01141	119.4
Markov chain ($B = 40$, $C = 10{,}000$)	—	.2167	.007824	238.9

Example 3. The Rhesus data shown in Figure 3 were reconstructed from Cavalli-Sforza and Bodmer (1971, pp. 224–228), conditioning on the reported phenotypic counts and estimated allele frequencies. For convenience, the alternative Rhesus notations are given in Table 4. Since the total sample size is extremely large (n = 8,297), it is clear that the conventional Monte Carlo method is not appropriate in this case. The results of goodness-of-fit and exact tests are shown in Table 5. Again, the P-values calculated from χ^2 with continuity corrections are erroneously small. It should be noted that, even for this large total sample size, employment of different goodness-of-fit test statistics could lead to completely different conclusions.

A_1	1,236							
A_2	120	3						
A_3	18	0	0					
A_4	982	55	7	249				
A_5	32	1	0	12	0			
A_6	2,582	132	20	1,162	29	1,312		
A_7	6	0	0	4	0	4	0	
A_8	2	0	0	0	0	0	0	0
A_9	115	5	2	53	1	149	0	0	4
	A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8	A_9

Figure 3. Genotype frequency data at Rhesus locus (Cavalli-Sforza and Bodmer, 1971).

Table 4
Alleles and haplotypes at Rhesus locus

Allele	Haplotype	Allele
A_1	cde	r
A_2	cDe	Ro
A_3	cdE	r″
A_4	cDE	R_2
A_5	Cde	r′
A_6	CDe	R_1
A_7	CDE	R_z
A_8	C^{w}de	r′^{w}
A_9	C^{w}De	R$_1^w$

Table 5
Results of goodness-of-fit and exact tests for Rhesus data. The dememorization period for Markov chain method is 1,000.

Statistic/Method	Value	*P*-value	S.E.	CPU time
F^2	21.8925	.9691	—	—
G^2	25.3359	.9078	—	—
χ^2	23.0401	.9536	—	—
$\chi^2_{.5}$	2,370.6736	<.0001	—	—
$\chi^2_{.25}$	604.2268	<.0001	—	—
Markov chain (B = 100, C = 1,000)	—	.7187	.02777	68.1
Markov chain (B = 100, C = 5,000)	—	.6955	.01666	337.2

6. Discussion

Our purpose in giving examples in Section 5 is twofold: to illustrate the desirability of using the exact test and to demonstrate the practicality of the method proposed here. It is not our intention to replace large-sample goodness-of-fit tests for testing HWP, but we encourage the use of the exact test whenever the sample size is small and/or the table in sparse, or simply when the adequacy of applying asymptotic results is in doubt.

The two proposed methods are computationally simple and can be implemented by very compact programs. The conventional Monte Carlo method has the advantage that the sample size of the simulation can be determined in advance to ensure that the estimation error is within some range with, say, 99% confidence, regardless of the size of the table. In contrast, the sample size of the simulation necessary to achieve desired accuracy for the Markov chain method increases with the size of the table. However, the Markov chain method has the advantage that it does not require the HWP probability of each newly generated table (nor even of the observed one) and that the time spent on each switch is virtually independent of the size of the table. Hence it can be used on any occasion, though it is most useful when the table is large and sparse. It is faster than the conventional Monte Carlo method when the sample size is moderate or large. For the conventional Monte Carlo, the time spent on generating a new table and on computing the corresponding probability is proportional to the sample size. It is insensitive to the number of alleles. Thus the conventional Monte Carlo method is most suitable for data with a large number of alleles but a small sample size.

For the conventional Monte Carlo method, a simulation size of 17,000 is usually sufficient when the true *P*-value is well above or below the prespecified significance level (see Section 3). For both methods, we suggest using the batch estimate of standard error to gauge the variability of the estimate. Greater accuracy and reliability are always available at extra computing cost.

For the Markov chain method, there is no simple way to determine how long the dememorization period should be. Basically, if the sample size or the number of alleles is large, one should use a longer period because there are more tables with the same margins as that of the observed table. Ideally, the longer the period is, the better the result will be, but an excessively long dememorization period may mean a lot of wasted computing time. In most cases, 1,000 steps seems to suffice, but larger problems will definitely need a longer period.

It should be pointed out that if a population does exhibit HWP it should not be taken as evidence that the conditions of the law are being satisfied in that population (Li, 1988; Hernández and Weir, 1989). Several of the assumptions of Hardy–Weinberg equilibrium may be violated simultaneously in such a way that their effects cancel each other.

ACKNOWLEDGEMENTS

This work was supported in part by NIH Grant NHLBI HL3 0086 and NSF Grant BSR-8619760. The authors are grateful to Joe Felsenstein for suggesting the conventional Monte Carlo algorithm, Ellen Wijsman for helpful discussions, and Charlie Geyer for sharing his expert knowledge in C and LaTeX. The authors also thank the two anonymous referees for their useful comments. The distribution of the computer program is supported by NIH Grant HG-00209.

RÉSUMÉ

La loi de Hardy–Weinberg joue un rôle important dans le domaine de la génétique des populations et sert souvent de base pour l'inférence génétique. Compte tenu de son importance, on a consacré beaucoup d'attention aux tests des proportions de Hardy–Weinberg (HWP) ces dernières dizaines d'années. On sait depuis longtemps que les tests d'ajustement asymptotique peuvent conduire à des résultats erronés quand la taille de l'échantillon et/ou les fréquences genotypiques sont petites. Un algorithme d'énumération complète pour un test exact a été proposé, mais on ne peut pas l'utiliser pour des loci ayant plus que quelques allèles à cause du volume de calcul. Nous proposons deux algorithmes pour estimer le niveau de signification pour un test de HWP. Les algorithmes sont facilement applicables à des loci avec des allèles multiples. Les deux sont remarquablement simples et de calcul rapide. On compare l'efficacité et les mérites des deux algorithmes. On donne des recommandations. On donne des exemples numériques pour illustrer la praticabilité des algorithmes.

REFERENCES

Agresti, A., Wackerly, D., and Boyett, J. M. (1979). Exact conditional tests for cross-classification: Approximation of attained significance levels. *Psychometrika* **44,** 75–83.

Becker, R. A., Chambers, J. M., and Wilks, A. R. (1988). *The New S Language: A Programming Environment for Data Analysis and Graphics.* Belmont, California: Wadsworth and Brooks/Cole.

Besag, J. and Clifford, P. (1989). Generalized Monte Carlo significance tests. *Biometrika* **76,** 633–642.

Binder, K. and Heermann, D. W. (1988). *Monte Carlo Methods in Statistical Physics.* Berlin: Springer-Verlag.

Botstein, D., White, R. L., Skolick, M., and Davis, R. W. (1980). Construction of a genetic linkage map in man using restriction fragment length polymorphism. *American Journal of Human Genetics* **32,** 314–331.

Cavalli-Sforza, L. L. and Bodmer, W. F. (1971). *The Genetics of Human Populations.* San Francisco: W. H. Freeman.

Chapco, W. (1976). An exact test of the Hardy–Weinberg law. *Biometrics* **32,** 183–189.

Crow, J. E. (1988). Eighty years ago: The beginnings of population genetics. *Genetics* **119,** 473–476.

Elston, R. C. and Forthofer, R. (1977). Testing for Hardy–Weinberg equilibrium in small samples. *Biometrics* **33,** 536–542.

Emigh, T. H. (1980). A comparison of tests for Hardy–Weinberg equilibrium. *Biometrics* **36,** 627–642.

Emigh, T. H. and Kempthorne, O. (1975). A note on goodness-of-fit of a population to Hardy–Weinberg structure. *American Journal of Human Genetics* **27,** 778–783.

Haldane, J. B. S. (1954). An exact test for randomness of mating. *Journal of Genetics* **52,** 631–635.

Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications. *Biometrika* **57,** 97–109.

Hernández, J. L. and Weir, B. S. (1989). A disequilibrium coefficient approach to Hardy–Weinberg testing. *Biometrics* **45,** 53–70.

Levene, H. (1949). On a matching problem arising in genetics. *Annals of Mathematical Statistics* **20,** 91–94.

Li, C. C. (1955). *Population Genetics.* Chicago: University of Chicago Press.

Li, C. C. (1988). Pseudo-random mating populations. In celebration of the 80th anniversary of the Hardy–Weinberg law. *Genetics* **119,** 731–737.

Louis, E. J. and Dempster, E. R. (1987). An exact test for Hardy–Weinberg and multiple alleles. *Biometrics* **43,** 805–811.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953). Equations of state calculations by fast computing machines. *Journal of Chemical Physics* **21,** 1087–1092.

Nakamura, Y., Leppert, M., O'Connell, P., Wolff, R., Hom, T., Culver, M., Martin, C., Fujimoto, E., Hoff, M., Kumlin, E., and White, R. (1987). Variable number of tandem repeat (VNTR) markers for human gene mapping. *Science* **2351,** 1616–1622.

Ripley, B. D. (1987). *Stochastic Simulation.* New York: Wiley.

Smith, C. A. B. (1986). Chi-squared tests with small numbers. *Annals of Human Genetics* **50,** 163–167.

Received February 1990; *revised October* 1990; *accepted February* 1991.

Appendix

Proof of Irreducibility of the Markov Chain of Section 4

We prove the result by induction.

For $m = 2$, the result is trivial. Now we assume it is true for $m' < m$. Consider the case of m alleles. Suppose $\mathbf{f}, \mathbf{g} \in \Gamma$. For notational convenience, we will make no distinction between the cell and its count. We first pool the alleles A_1 and A_2 together, as if they were the same. Then we have two tables of order $m - 1$, say, $\mathbf{f}'$ and $\mathbf{g}'$, and, by assumption, $\mathbf{g}'$ can be obtained from $\mathbf{f}'$ through a sequence of switches. These switches can obviously be regarded as made on $\mathbf{f}$ so that, after switch, the 3rd, 4th, ..., mth columns of $\mathbf{f}$ and $\mathbf{g}$ are exactly the same. The only difference is the first two columns. We now prove that there exist some switches that do not touch the 3rd, 4th, ..., and mth columns and after these switches the two tables will be exactly the same.

Note that, because of marginal constraints, $f_{j1} + f_{j2} = g_{j1} + g_{j2}$ $(j = 3, 4, \ldots, m)$. Without loss of generality, we can assume $f_{m1} > g_{m1}$ since every switch is reversible. Now, suppose $k_m = f_{m1} - g_{m1} = g_{m2} - f_{m2} > 0$. Without loss of generality, we can also assume that $g_{22} \geq f_{22}$; otherwise we can choose $i_1 = 2$, $i_2 = m$ and $j_1 = 1, j_2 = 2$ and make $\min(f_{22} - g_{22}, k_m)$ R_0-switches so that, after these switches, either $f_{m1} = g_{m1}$ or $g_{22} = f_{22}$ or both. Now, since

$$\begin{aligned} f_{21} + f_{22} + \cdots + f_{m-1,2} &= f_{\cdot 2} - f_{22} - f_{m2} \\ &= (g_{\cdot 2} - f_{22} - g_{m2}) + k_m \\ &\geq (g_{\cdot 2} - g_{22} - g_{m2}) + k_m \quad (\text{as } g_{22} \geq f_{22}) \\ &\geq k_m, \end{aligned}$$

we can always find k_m R_0-switches (all with $j_1 = 1$, $j_2 = 2$, $i_2 = m$, but with varying i_1), so that, after these switches are performed, $f_{m1} = g_{m1}$ and $f_{m2} = g_{m2}$. Note that the switch involves only the first and second columns.

By the same token, we can successively find k_i switches so that, after these switches, $f_{i1} = g_{i1}$ and $f_{i2} = g_{i2}$ $(i = m - 1, m - 2, \ldots, 4, 3)$, without touching either 3rd, 4th, ..., mth columns or cells in rows below the ith. After the case $i = 3$, the only difference is f_{11}, f_{21}, and f_{22}. Again, due to the marginal constraints, $2f_{11} + f_{21} = 2g_{11} + g_{21}$, $2f_{22} + f_{21} = 2g_{22} + g_{21}$. This is the case of $m = 2$. We can find $|f_{11} - g_{11}|$ D_2-switches (if $f_{11} \geq g_{11}$) or R_2-switches (if $f_{11} \leq g_{11}$) so that $f_{11} = g_{11}$, $f_{21} = g_{21}$, and $f_{22} = g_{22}$ after these switches. Thus we complete our proof.