Received: 14 September 2020 | Revised: 24 February 2021 | Accepted: 1 March 2021

DOI: 10.1111/1755-0998.13373

MOLECULAR ECOLOGY RESOURCES WILEY

RESOURCE ARTICLE

A network algorithm for the X chromosomal exact test for Hardy–Weinberg equilibrium with multiple alleles

Jan Graffelman[1,2] | Leonardo Ortoleva[1,3]

[1]Department of Statistics and Operations Research, Universitat Politècnica de Catalunya, Barcelona, Spain

[2]Department of Biostatistics, University of Washington, Seattle, WA, USA

[3]Department of Control and Computer Engineering, Politecnico di Torino, Torino, Italy

Correspondence
Jan Graffelman, Department of Statistics and Operations Research, Universitat Politècnica de Catalunya, Carrer Jordi Girona 1-3, 08034 Barcelona, Spain.
Email: jan.graffelman@upc.edu

Funding information
Spanish Ministry of Science, Innovation and Universities, Grant/Award Number: RTI2018-095518-B-C22; European Regional Development Fund; United States National Institutes of Health, Grant/Award Number: R01 GM075091

Abstract

Statistical methodology for testing the Hardy–Weinberg equilibrium at X chromosomal variants has recently experienced considerable development. Up to a few years ago, testing X chromosomal variants for equilibrium was basically done by applying autosomal test procedures to females only. At present, male alleles can be taken into account in asymptotic and exact test procedures for both the bi- and multiallelic case. However, current X chromosomal exact procedures for multiple alleles rely on a classical full enumeration algorithm and are computationally expensive, and in practice not feasible for more than three alleles. In this article, we extend the autosomal network algorithm for exact Hardy–Weinberg testing with multiple alleles to the X chromosome, achieving considerable reduction in computation times for multiallelic variants with up to five alleles. The performance of the X chromosomal network algorithm is assessed in a simulation study. Beyond four alleles, a permutation test is, in general, the more feasible approach. A detailed description of the algorithm is given, and examples of X chromosomal indels and microsatellites are discussed.

KEYWORDS
Hardy–Weinberg equilibrium, indel, microsatellite, network algorithm, permutation test, X chromosome

1 | INTRODUCTION

The statistical testing of genetic variants for the Hardy–Weinberg equilibrium (HWE) is an important part of the analysis of genetic data sets, for a variety of reasons. Gross deviations from equilibrium are often the result of genotyping errors, and testing can be helpful to detect such errors (Chen et al., 2017; Hosking et al., 2004; Leal, 2005; Teo et al., 2007). Moreover, many methods used in genetic data analysis rely on the equilibrium assumption, and the filtering of variants on the basis of their *p*-values obtained in a test for HWE can be used as a safeguard to prevent violation of assumptions made. A recent overview of statistical tests for the Hardy–Weinberg equilibrium is given by Graffelman (2020). Currently, exact test procedures are the state of the art for testing biallelic genetic variants and are most commonly employed. Fast recursive procedures are available that can do exact testing of biallelic variants for HWE on a genome-wide scale (Chang et al., 2015; Wigginton et al., 2005). For variants with multiple alleles, the exact test is computationally more costly. Algorithms for the efficient exact testing of multiallelic variants have been proposed by several authors (Guo & Thompson, 1992; Huber et al., 2006; Louis & Dempster, 1987). A recursive network algorithm (Aoki, 2003; Engels, 2009) has been proposed for more efficient calculation of exact *p*-values. When the computational cost of the network approach becomes prohibitive, a permutation test based on the sampling of outcomes from the exact

distribution can still be used as an alternative in order to obtain an approximate *p*-value (Guo & Thompson, 1992).

Recently, the statistical testing for HWE of variants on the X chromosome has experienced considerable development. Up to a few years ago, X chromosomal variants were tested using autosomal procedures for the females only. Graffelman and Weir (2016) developed the full suite of frequentist test procedures (a two degrees of freedom asymptotic chi-square test, an exact test and a permutation test) specifically for X chromosomal variants, which take male alleles on X into account. This has the advantage of a larger sample size (more X chromosomes), implying higher precision for the estimation of allele frequencies, and the potential rejection of equilibrium when there is a difference in allele frequency between the sexes. The biallelic exact test for X is computationally feasible for a complete X chromosome, and efficient C++ code for this test is currently shared by the PLINK software (Chang et al., 2015; Purcell et al., 2007) and the R-package HardyWeinberg (Graffelman, 2015). Later, Graffelman and Weir (2018) extended their X chromosomal exact test for multiple alleles with a classical full enumeration algorithm, and reported on the analysis of all triallelic variants on X at considerable computational cost, and suggesting the use of permutation tests based on sampling for X chromosomal variants with four or more alleles. In this article, we propose a modification of the network algorithms proposed by Aoki (2003) and Engels (2009), adapting the network algorithm for the X chromosome. The network algorithm efficiently avoids the repeated calculation of factorial terms that are shared in the list of possible outcomes generated by complete enumeration, leading to large computational savings. This way, we strive to extend the application of the X chromosomal exact test towards variants with a larger number of alleles while maintaining computation time within feasible limits.

The structure of the remainder of this article is as follows. In Section 2, we review exact tests with multiple alleles for the autosomes and for the X chromosome, and present the adaptation of the network algorithm to the X chromosome. In Section 3, we assess the performance of the new network algorithm in a simulation study. Section 4 shows the examples of the network-based test for a varying number of alleles with data taken from the 1,000 genomes project (The 1,000 Genomes Project Consortium, 2015). We describe the HWE analysis of a complete X chromosome of the Tuscan population (TSI) of the 1,000 Genomes Project and also address the analysis of a forensic database of X chromosomal microsatellites (Chen et al., 2018). The Discussion in Section 5 completes the manuscript.

2 | THEORY

In this section, we review exact inference for HWE with multiple alleles and explain the operation of the network algorithm for both the autosomal and X chromosomal cases with a toy example.

2.1 | Autosomal exact inference with multiple alleles

Exact inference for autosomal variants with multiple alleles is based on the conditional distribution of the genotype counts, considering all observed allele counts as given. This distribution was derived by Levene (1949) and is given by

$$P(N_{ij} = n_{ij} \mid n_1, \ldots, n_k) = \frac{n!\, 2^{n-d} \prod_{i=1}^{k} n_i!}{(2n)! \prod_{i \geq j} n_{ij}!}, \tag{1}$$

where n represents the sample size; n_i, the count of the i th allele; n_{ij}, the count of genotype ij; and $d = \sum n_{ii}$, the total homozygote frequency. Equation (1) describes the distribution of the genotype counts under the assumption of HWE. One first calculates the probability of the observed sample according to Eq. (1). Next, a full enumeration is made of all possible genotype arrays that are compatible with the observed total allele counts, and their probabilities are calculated. Finally, the exact *p*-value is obtained by summing the probabilities of all genotype arrays that are less likely than or equally likely to the observed sample. The full enumeration approach combined with the calculation of Eq. (1) is computationally expensive, in particular for large samples with many alleles. A full enumeration algorithm for an arbitrary number of alleles has been described by Louis and Dempster (1987).

A drawback of the classical full enumeration algorithm is that many genotype arrays involve the same factorials, which will be repeatedly calculated if a simple loop is used to iterate over all possible arrays. The network algorithm enables the sharing of the calculation of common factorials across similar genotype arrays and can so produce considerable computational savings. The network approach was proposed by Mehta and Patel (1983) who developed this algorithm for a more efficient calculation of Fisher's exact test for large contingency tables. Aoki (2003) presented the first network algorithm for exact testing in the context of Hardy–Weinberg equilibrium. The computation of Eq. (1) is simplified by recognizing that for given allele counts, the part

$$K_p = \ln\left(\frac{n!\, 2^n \prod_{i=1}^{k} n_i!}{(2n)!}\right), \tag{2}$$

is a common factor for all genotype arrays and can be taken as a constant, which is calculated only once. To further simplify the calculations, we take the logarithm of Eq. (1) and have

$$\ln\left(P\left(N_{ij}\right)\right) = K_p - \sum_{i \geq j} \ln\left(n_{ij}!\right) - d \ln(2). \tag{3}$$

Figure 1 shows the operation of an autosomal network algorithm for a triallelic variant, with a sample of size $n = 8$ and allele counts 7, 5 and 4 for A, B and C, respectively. Each path from left to right generates a particular genotype array. The probability of the sample generated by the traced path is 0.028, and if this sample is observed, the

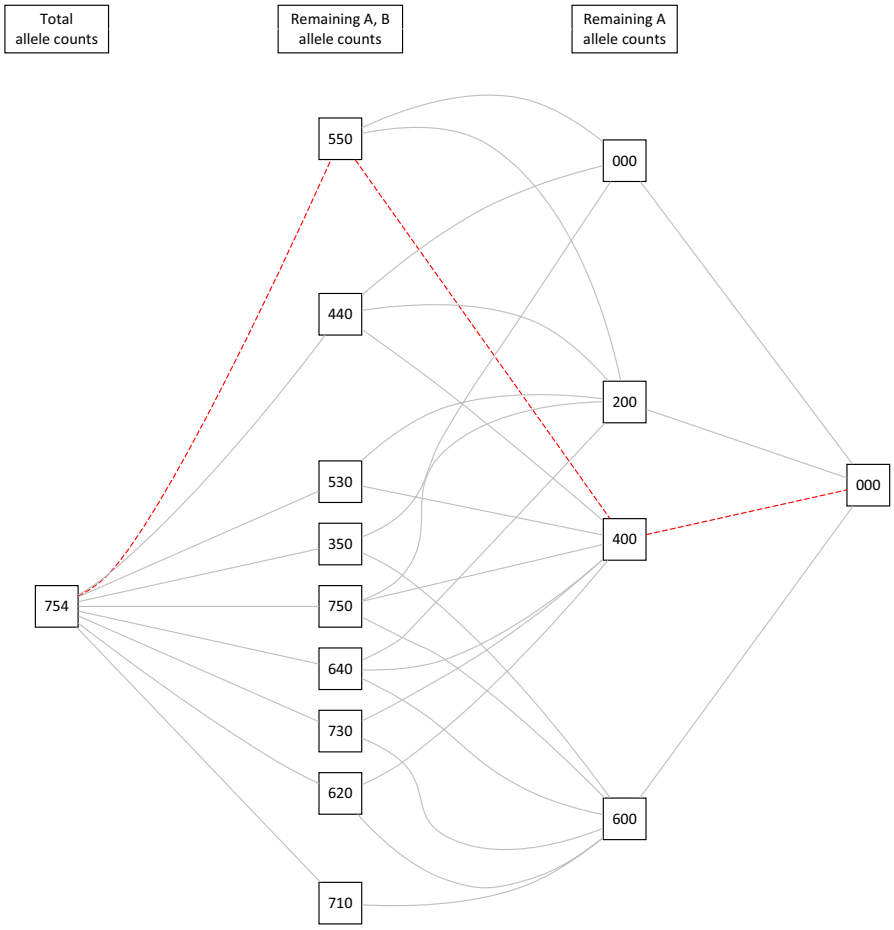

FIGURE 1 Graph for an autosomal triallelic variant for a sample size of $n = 8$ with allele counts (A = 7, B = 5, C = 4). Nodes represent allele counts, and edges, the assignment of alleles to genotypes. The dashed path illustrates the generation of 1 CC homozygote and 2 AC heterozygotes, leaving allele counts (A = 5, B = 5, C = 0), followed by the generation of two BB homozygotes and one AB heterozygote, leaving (A = 4, B = 0, C = 0), and finally the generation of two AA homozygotes to arrive at (A = 0, B = 0, C = 0). The generated genotype array is (AA = 2, BB = 2, CC = 1, AB = 1, AC = 2, BC = 0). Each path in the network traces the generation of a genotype array that is compatible with the observed allele counts. The network exhausts all possible genotype arrays for the given allele counts

exact test p-value is 0.167. The network has 21 paths corresponding to 21 different possible genotype arrays for the given allele counts.

The algorithm proceeds by computing the second term of logfactorials in Eq. (3) incrementally, exhausting alleles one by one. As Figure 1 shows, the edge from 754 to 550 is shared by three genotype arrays, and the corresponding logfactorials of n_{AA} and n_{AC} only need to be computed once. For more details on the autosomal network algorithm, we refer to Aoki (2003) and Engels (2009).

2.1.1 | X chromosomal exact inference with multiple alleles

For exact testing for Hardy–Weinberg equilibrium at X chromosomal variants with multiple alleles, Graffelman and Weir (2018) derived, assuming equality of allele frequencies in the sexes and Hardy–Weinberg proportions in females, the exact joint distribution of the number of female heterozygotes and the number of hemizygous males given by

$$P\left(N_{fij} = n_{fij} \cap N_{mi} = n_{mi} | n_1, \ldots, n_k\right) = \frac{n_m!\, n_f!\, 2^{n_f - d} \prod_{i=1}^{k} n_i!}{n_t!\, \prod_{i=1}^{k} n_{mi}! \prod_{i \geq j} n_{fij}!} \quad (4)$$

where n_m and n_f represent the numbers of males and females; $n_t = 2n_f + n_m$, the total number of alleles; n_{mi} and $n_{fij,}$ male and female genotype counts; and d, the total number of homozygote females. To show the increase in computational complexity, we use the same set of allele counts (A = 7, B = 5, C = 4) as in Figure 1, but now consider gender, assuming the sample is composed of 4 males and 6 females, totalling 16 alleles. Figure 2 shows the network for this case, and the construction of the genotype array $(m_A = 3, m_B = 1, m_C = 0, f_{AA} = 1, f_{BB} = 2, f_{CC} = 1, f_{AB} = 0, f_{AC} = 2, f_{BC} = 0)$ is indicated. The number of possible genotype arrays, 136, has increased considerably in comparison with the previous autosomal variant with the same total allele counts. We follow the same approach as before, now defining two constants K_p and K_m as

$$K_p = \ln\left(\frac{n_m!\, n_f!\, \prod_{i=1}^{k} n_i!}{n_t!}\right) \quad \text{and} \quad K_m = \ln\left(\frac{2^{n_f}}{\prod_{i=1}^{k} n_{mi}!}\right). \quad (5)$$

Taking logarithms, one has

$$\ln\left(P\left(N_{ij}\right)\right) = K_p + K_m - \sum_{i \geq j} \ln\left(n_{fij}!\right) - d\ln(2), \quad (6)$$

and again, the sum of the logfactorials is incrementally evaluated one allele at a time. In essence, for X chromosomal variants we first generate all possible male genotype arrays and next apply the autosomal network algorithm using the remaining female allele counts.

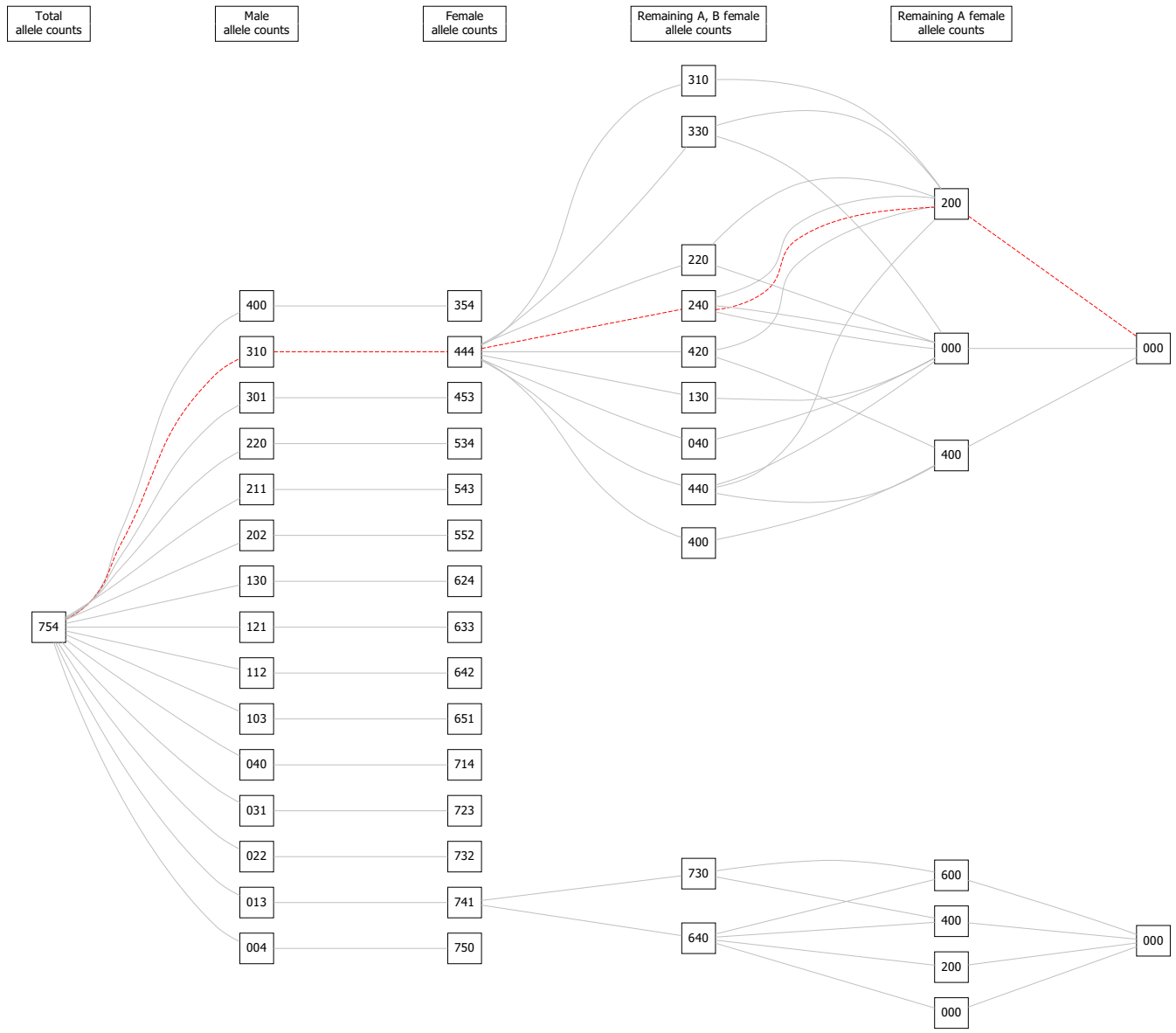

FIGURE 2 Network for an X chromosomal triallelic variant for a sample size of 4 males and 6 females with total allele counts (A = 7, B = 5, C = 4). Nodes represent allele counts, and edges, the assignation of alleles to genotypes. The second column of the network shows all possible male allele counts given the available alleles and given that there are 4 males. The third column gives the allele counts available for females, once male allele counts have been subtracted from the total. Fourth, fifth and sixth columns give the remaining female allele counts after assigning C, B and A female alleles, respectively. The dashed path illustrates the generation of 3 A males and 1 B male, followed by the generation of the females. The generated genotype array is ($m_A = 3, m_B = 1, m_C = 0, f_{AA} = 1, f_{BB} = 2, f_{CC} = 1, f_{AB} = 0, f_{AC} = 2, f_{BC} = 0$). Each path in the network traces the generation of a genotype array compatible with the observed allele counts. The network exhausts all possible genotype arrays for the given allele counts and the given number of males and females. For simplicity, the network for female genotypes is shown only for two sets of female allele counts

3 | SIMULATION STUDY

The X chromosomal exact test involves the computation of the probabilities of all possible genotype arrays for the given allele counts, according to Eq. (4). The number of arrays is, in general, larger than for the autosomes, and the X chromosomal exact test is computationally expensive for systems with multiple alleles. We compare the computational cost of the network algorithm with a classical enumeration algorithm for bi- and triallelic variants, and also compare the network algorithm with a permutation test for three or more alleles. We expect the network algorithm to be computationally cheaper because it is able to store partial results thanks to recursion through the network, which avoids repeating calculations from the beginning for every possible table of genotypes under analysis, as explained in the previous section. Full enumeration algorithms for X chromosomal exact test are currently only available for bi- and triallelic variants. Figure 3a shows the computation time as a function of the number of biallelic X

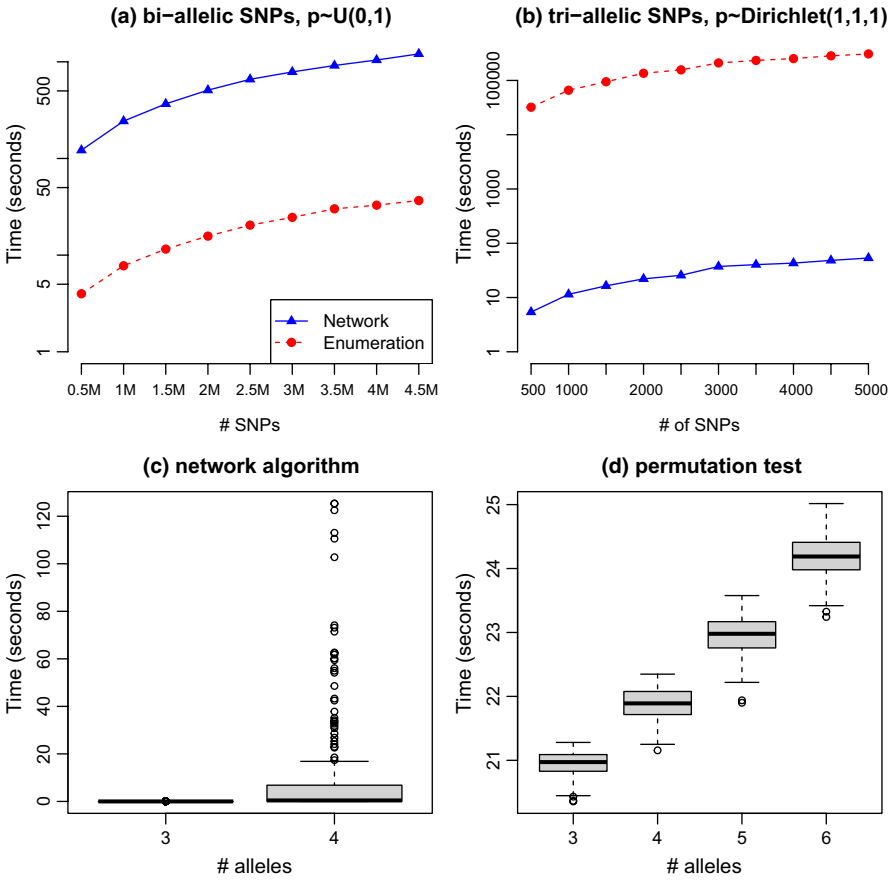

FIGURE 3 Execution times in seconds for the classical enumeration algorithm, the network algorithm and the permutation test. (a) Execution time (in a logarithmic scale) as a function of the number of biallelic SNPs with uniform allele frequencies. (b) Execution time (in a logarithmic scale) as a function of the number of triallelic SNPs with the Dirichlet (1, 1, 1) allele frequencies. (c) Box plot of execution times of the network algorithm for 250 SNPs with three and four alleles. (d) Execution times of the permutation test for 250 SNPs with three, four, five or six alleles

chromosomal SNPs. X chromosomal SNPs were simulated under the assumptions of Hardy–Weinberg proportions in females and equality of male and female allele frequencies, using a sample size of $n = 100$. For example, biallelic X chromosomal SNPs were simulated by drawing samples from a multinomial distribution with probability vector $\left(\frac{1}{2}p, \frac{1}{2}q, \frac{1}{2}p^2, pq, \frac{1}{2}q^2\right)$. All computations were carried out in the R environment (R Core Team, 2020), using a server with thirty-two compute nodes, half of the nodes were 16-core Intel Xeon E5-2630 Systems (2.40 GHz; 128 Gb RAM); the other half were 24-core Intel Xeon Gold 5118 (2.30 GHz; 384 Gb RAM). For two alleles, the network algorithm is seen to take more time in comparison with an enumeration algorithm. The computation time of the network algorithm is seen to increase, as expected, linearly with the number of SNPs, though most conveniently shown in a logarithmic scale as in Figure 3a,b. We simulated up to 4.5 M biallelic SNPs, because the 1,000 Genomes Project (The 1,000 Genomes Project Consortium, 2015) reports about 3.5 M biallelic SNPs on X. For biallelic SNPs, we used the HWExactStats implementation of the enumeration algorithm and the HWNetwork implementation of the network algorithm. The first uses C code shared with PLINK, and the latter modified C code of Engels' autosomal algorithm. We also compared the actual results (the *p*-values) of the network algorithm and the enumeration algorithm. For biallelic SNPs, we found a very good agreement between *p*-values obtained with the enumeration algorithm and the network algorithm. The largest difference between the *p*-values of the two algorithms was as small as $2.62e^{-13}$. The theoretical expectation is, as the data are simulated under the equilibrium hypothesis, that at a 5 per cent significance level, about 5 per cent significant results will be observed. In this sense, for the simulations with 10,000 biallelic variants we obtained a rejection rate of 4.55 per cent, close to the theoretically expected rate.

These calculations were repeated for simulated triallelic variants, for which the results are shown in Figure 3b, where we simulated up to 5,000 variants, which is close to the amount of triallelics found on X in the 1,000 Genomes Project (see Figure 4a). These figures show that for triallelic variants the network algorithm is much faster than the enumeration algorithm. We note that it takes the enumeration algorithm 85.5 hours to calculate the maximum of 5,000 X-triallelics, whereas the network algorithm does this in 53.4 seconds. Execution times also increase linearly with the number of SNPs. For larger numbers of alleles, an enumeration algorithm is currently not available. For three through six alleles, we compare the network algorithm with the permutation test. We generated 250 multiallelic polymorphisms for a given number of alleles under the assumption of equal allele frequencies in the sexes and Hardy–Weinberg proportions for females, using the Dirichlet distribution with all concentration parameters equal to 1 to simulate the allele frequencies. Figure 3c,d shows box plots of the execution time (in seconds) for the 250 simulated variants as a function of the number of alleles for both the network and permutation tests. The execution time of the permutation tests

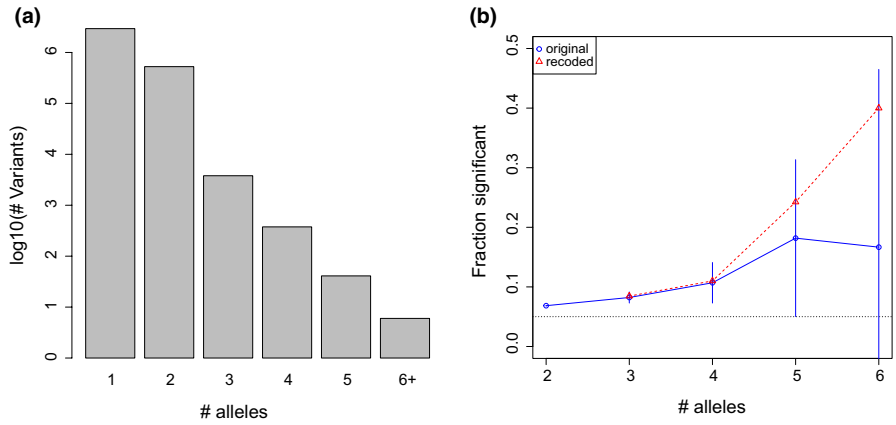

FIGURE 4 (a) Number of variants with a given number of alleles for the TSI population. (b) Fraction of significant variants for a given number of alleles. Vertical lines represent 95% confidence intervals for the theoretical fraction. The horizontal dotted reference line represents the significance level $\alpha = 0.05$. Blue open dots represent observed fractions of significant variants. Red open triangles present observed fractions of significant variants when the polymorphism is recoded as biallelic

only experiments a minor increase when the number of alleles increases from three to four. For three and four alleles, the network algorithm generally provided the fastest solution. For four alleles, Figure 3c shows some hard polymorphisms appear for which the network needs more time than a permutation test. On average, the network algorithm is much faster and outperforms the permutation test with 17,000 draws. Beyond four alleles, the permutation test is feasible for all polymorphisms, whereas the computational cost of the network algorithm becomes prohibitive.

4 | EMPIRICAL DATA EXAMPLES

We present examples of the application of X chromosomal exact tests based on the network algorithm for multiallelic variants taken from the 1,000 Genomes Project (The 1,000 Genomes Project Consortium, 2015) and for a forensic database of X chromosomal microsatellites (Chen et al., 2018).

4.1 | TSI sample of the 1,000 Genomes Project

We consider the analysis of a complete X chromosome of a sample of the TSI population (Tuscany, Italy) of the 1,000 Genomes Project, using all its multiallelic variants stored in the VCF files of the project, and using the VCFR package (Knaus & Grünwald, 2017) to process the data. This data set consists of 107 individuals, 53 males and 54 females. Variants in the pseudo-autosomal regions (Graves et al., 1998) were excluded from the analysis. Figure 4a shows a bar plot with the prevalence of variants with a given number of alleles and confirms the well-known fact that for a given human population, most variants are monomorphic or biallelic. We calculated the fraction of significant variants (using $\alpha = 0.05$) for each given number of alleles, which reveals that for multiallelic variants more evidence for disequilibrium is found, as shown in Figure 4b.

Using the enumeration algorithm for all biallelic X chromosomal variants, the network algorithm for all variants with three through five alleles and the permutation test to analyse variants with six or more alleles, it took about 10 minutes to analyse all polymorphisms of the TSI sample ($n = 107$); this could be reduced if a few hard three through five allelic variants would be resolved by using the permutation test, at the expense of less precision. We illustrate the observed faster computation of X chromosomal exact test results with some triallelic polymorphisms. Table 1 shows genotype counts and execution times for six different SNPs. Enumeration and network algorithm produce the same p-value, and the permutation p-value is close to these p-values. For 17,000 permutations, which are needed to estimate the p-value with a precision of 0.01 (Ziegler & König, 2006, Chapter 4], the permutation test takes about half a minute to complete. The enumeration algorithm is faster than the permutation test for those variants that have a dominant major allele. For variants rs200225892 and rs11439044, alternate alleles have substantial counts, and in these cases, the permutation test is faster than full enumeration. In all cases, the network algorithm outperforms the permutation and enumeration tests. The network algorithm also requests more computation time for the two variants with larger alternate allele frequencies.

Interpreting the genotype patterns, one sees that for rs369254025 HWE is rejected because of different allele frequencies for the sexes and excess heterozygosity for females; for rs56005969, no significant deviations are found; for rs185941206, HWE is rejected because females are monomorphic, whereas males carry all three alleles; for rs200225892, all three alleles are common and no significant deviations are found; for rs11439044, females are out of HW proportions; and finally, for rs112679846 males are monomorphic, but females have a large number of alternate alleles. Notice that disequilibrium would have gone unnoticed for variants rs185941206 and rs112679846 if equilibrium would have been tested in females only. SNP rs58533540 is, according to the exact test significant at a usual significance level of five per cent, though

TABLE 1 Genotype counts, *p*-values and execution times (in seconds) for permutation, enumeration and network algorithm of six SNPs of the TSI sample of the 1,000 Genomes Project. Exact *p*-values for a test of equality of allele frequencies (EAF) and HWP in females only (FO) are also reported

	Males			Females						*p*-values			Execution time			*p*-values	
SNP id	A	B	C	AA	AB	AC	BB	BC	CC	Perm.	Enum.	Netw.	Perm.	Enum.	Netw.	EAF	FO
rs369254025	46	1	6	22	0	31	0	0	1	0.0011	0.0006	0.0006	24.42	0.16	0.02	0.0053	0.0113
rs56005969	52	1	0	50	0	4	0	0	0	0.2134	0.2091	0.2091	31.37	0.00	0.00	0.1718	1.0000
rs185941206	50	2	1	54	0	0	0	0	0	0.0450	0.0469	0.0469	26.53	0.00	0.00	0.0343	1.0000
rs200225892	20	19	14	9	16	8	3	13	5	0.5006	0.5000	0.5000	28.90	478.29	0.13	0.9325	0.2362
rs11439044	18	22	13	7	7	17	1	18	4	0.0104	0.0119	0.0119	27.05	494.50	0.13	0.0587	0.0257
rs112679846	53	0	0	38	15	1	0	0	0	0.0055	0.0048	0.0048	26.22	0.03	0.00	0.0027	0.6344
rs58533540	15	37	1	4	42	0	8	0	0	0.0000	1.7E−05	1.7E−05	21.43	0.35	0.00	0.0241	0.0001

not significant if a genome-wide significance level of $5 \cdot 10^{-8}$ is employed, as if often used in large-scale association studies in order to correct for multiple testing (Fadista et al., 2016; Panagiotou et al., 2012; Roeder & Wasserman, 2009; Xu et al., 2014). We note that the permutation test fails to correctly assess the significance of this variant at this level for not having sufficient precision (see Discussion).

4.2 | X chromosomal STRs of Han Chinese

We use a forensic database of 19 X-STRs of 206 unrelated Han Chinese individuals from Guizhou (104 females and 102 males) described by Chen et al., (2018). Table 2 gives the *p*-values of permutation tests and network algorithm exact tests for HWE along with the execution time. The X chromosomal exact test for all individuals was used, as well as an autosomal test that uses the females only. On average, the X chromosomal permutation test with 17,000 draws takes about 52 seconds to complete. We observe good agreement between the *p*-values obtained by the permutation test and by the network algorithm. The X chromosomal network algorithm is seen to be much faster for a four-allele STR, slower for a five-allele STR and not feasible for the remaining STRs which have 7+ alleles, for taking too much computation time.

The permutation test is, as expected, slightly faster for tests that use females only because of a smaller number of alleles. The application of the network algorithm to the females only leads to spectacular savings in computation time for the two STRs with four and five alleles; for seven or more alleles, the permutation test outperforms the network algorithm. Two STRs, DXS8378 and DXS10101, appear as significant at the 5% level; DX8378 for having different allele frequencies in the sexes ($p = 0.022$); and DXS10101 for having females out of Hardy–Weinberg proportions ($p = 0.008$).

5 | DISCUSSION

We have developed a network algorithm for the X chromosomal exact test for the Hardy–Weinberg equilibrium with multiple alleles. X chromosomal exact tests were hitherto only feasible for two or three alleles by using a classical full enumeration algorithm. For analysing variants with more alleles, a permutation test was required. The network algorithm proposed in this study extends the feasibility of the X chromosomal exact test. It is now possible to obtain exact *p*-values for triallelic X chromosomal variants within fractions of a second (see Table 1). In general, for variants with over four alleles, the computational cost of the network algorithm is still prohibitive, and one still needs to resort to a permutation test or Markov chain approach to resolve these cases. The current implementation of the X chromosomal network algorithm is based on Engels' autosomal network algorithm (Engels, 2009), which is still based on exhaustive listing of all tables. We expect that further computational savings can be achieved by trimming paths in the network (Aoki, 2003). In principle, exact *p*-values

TABLE 2 Test results and execution times for X chromosomal STRs. STR identifier, number of STR alleles, permutation test *p*-value and execution time, network-based exact test *p*-value and execution time, and the same test results based on an autosomal test for the females only, for 19 X-STRs. Dashes (–) represent results not available for requiring too much computation time. Execution times are expressed in seconds (s), minutes (m) or hours (h) as convenient

			All individuals				Females only			
			Permutation		Network		Permutation		Network	
	STR	No. alleles	*p*-value	Time	*p*-value open dots represent obser	Time (s)	*p*-value	Time (s)	*p*-value	Time (s)
1	DXS8378	5	0.0355	42 s	0.0345	951	0.0548	40 s	0.0545	0.012 s
2	DXS7423	4	0.5171	40 s	0.5099	0.3	0.2678	39 s	0.2628	0.003 s
3	DXS10148	17	0.4152	69 s	–	–	0.3905	67 s	–	–
4	DXS10159	10	0.0768	49 s	–	–	0.0846	48 s	–	–
5	DXS10134	16	0.3472	64 s	–	–	0.6156	62 s	–	–
6	DXS7424	10	0.9619	48 s	–	–	0.8616	47 s	0.8573	37.1 h
7	DXS10164	9	0.5226	46 s	–	–	0.3258	45 s	0.3238	77.0 s
8	DXS10162	10	0.6986	48 s	–	–	0.3981	47 s	0.4025	6.9 h
9	DXS7132	8	0.7882	45 s	–	–	0.5960	45 s	–	–
10	DXS10079	10	0.6003	49 s	–	–	0.8403	49 s	–	–
11	DXS6789	9	0.4776	47 s	–	–	0.7255	47 s	–	–
12	DXS101	12	0.1845	54 s	–	–	0.1572	53 s	–	–
13	DXS10103	7	0.8630	44 s	–	–	0.8957	44 s	–	–
14	DXS10101	19	0.0456	75 s	–	–	0.0082	73 s	–	–
15	HPRTB	8	0.7551	45 s	–	–	0.3819	44 s	0.3800	8.1 m
16	DXS6809	10	0.1722	49 s	–	–	0.1194	49 s	–	–
17	DXS10075	9	0.4762	46 s	–	–	0.3136	46 s	0.3109	72.2 m
18	DXS10074	11	0.6746	50 s	–	–	0.1791	49 s	–	–
19	DXS10135	21	0.5691	83 s	–	–	0.1308	82 s	–	–

are preferable over permutation *p*-values for giving an exact answer. In exact tests with discrete count data, such as the exact test for HWE, the *p*-value of the test can be defined in different ways (Graffelman & Moreno, 2013, Figure 1). The standard way to calculate the *p*-value is to sum the probabilities of all possible outcomes that are as likely or less likely as the observed data. In the context of HWE, using this standard *p*-value is known to be conservative (Wigginton et al., 2005). Graffelman and Moreno (2013) advocated the use of the *mid p*-value in exact tests for HWE, in a biallelic setting, for having a rejection rate that is closer to the nominal significance level. In this study, in the current multiallelic setting, we have used the standard *p*-value; the mid *p*-value can easily be obtained by subtracting half the probability of the observed sample, using Eqs. (1) and (4) for the autosomal and X chromosomal case, respectively, from the standard exact *p*-value obtained by the network or the permutation algorithm.

In modern genetic studies, a genome-wide significance level of $\alpha = 5 \cdot 10^{-8}$ is often employed in order to correct for multiple testing. Assessing significance at such a threshold with a precision of 10^{-8} would require over 10^{16} permutations, which is computationally not feasible, and this clearly emphasizes the need for obtaining exact *p*-values. For example, SNP rs58533540 in Table 1 has an exact *p*-value of $1.7 \cdot 10^{-5}$ and is not significant at the threshold $\alpha = 5 \cdot 10^{-8}$. The *p*-value of the permutation test obtained for this variant is 0, because none of the 17,000 (by default) permuted genotype tables had a probability below that of the observed table. The permutation test suggests the variant to be significant, but in fact the test is not able to assess the significance at the given genome-wide level, or would only be able to do so at an astronomical computational cost. X chromosomal STRs with over four alleles are common, and it remains a challenge to further improve algorithms for obtaining exact instead of approximate *p*-values in this setting. In forensics, where STRs with many alleles are widely used, a permutation test and Markov chain algorithm thus remain the best general purpose methods that will serve for all STRs. The network algorithm may be more interesting for the analysis of indels (Mills et al., 2006), which have in general a much smaller number of alleles. The execution times of the network algorithm can vary considerably for variants with the same number of alleles (see Figure 3c). For example, STRs DXS10162, DXS7424 and DXS10159 all have ten alleles but take 6.9, 37.1 and beyond 37.1 hours to compute. The particular set of allele counts will determine the complexity of the network and its computational cost.

The analysis of the complexity of the algorithms to use is certainly one of the key factors to study, in order to understand whether a new solution can achieve a certain type of efficiency in relation to the existing ones. For this reason, we want to briefly sort out the reasons that confirm theoretically why the new algorithm is faster than the old one. Looking at the classical full enumeration algorithm from this point of view, because of the two nested cycles, that allow to enumerate a priori all possible genotype matrices obtainable from the total number of alleles, it follows a quadratic complexity. The calculations are made using the matrices obtained in a linear way. Therefore, we can conclude that the classical algorithm for the HWE test follows an $O(n^2)$ complexity, since we take the worst-case scenario, which is always achieved regardless of the problem. About the new network algorithm, instead, there is a first recursion that goes to list all possible male individuals, constructing the vector of the alleles and analysing every possible case. This is in addition to the next recursion launched for female individuals that analyses at each iteration a vector of size always smaller by a factor of 1 compared with that of the previous iteration (movement from one column of the network to the next). Therefore, the trend, in this case, is logarithmic and does not result in the worst-case linear because computationally the results stored during the path are considered, so as not to start again to analyse each vector from the beginning. The combination of the two considerations can lead us to the conclusion of an algorithm with a complexity $O\ n\ \log(n)$. So, from a first theoretical analysis on the complexity of the two algorithms compared, since the network algorithm follows, in the worst case, the proportion just explained, it is more advantageous to use. In general, in computer science, when trying to find faster ways to solve certain problems, the price to pay is that of memory to be used. The new network algorithm assumes that the enumeration of the matrices of genotypes will be complete as for the classical algorithm, what changes is the way to exploit these data in the calculations. In order to achieve computational improvement, the proposed new approach takes full advantage of the strength of the recursive technique by building an ever-deepening stack of nested function calls. This certainly creates a complication from the point of view of the space used that the classical algorithm did not provide. To do this, it was decided to use the C language, more oriented to this approach, rather than R. To conclude, on the one hand a computationally better approach was fully exploited, on the other hand the best programming-level tool was chosen to put it into practice. The combination of the two makes the new approach faster than the previous one.

In exact testing for HWE with multiallelic genetic variants, *tied outcomes* can easily arise. A pair of tied outcomes refers to two different genotype arrays that have theoretically exactly the same probability under the equilibrium null distribution. Tied outcomes can be problematic if they involve the observed sample. If a genotyping array has the same probability as the observed sample, its probability should be included in the calculation of the *p*-value. Due to finite precision in the comparison of floating point numbers on a computer, a theoretically tied outcome may not be recognized as having the same probability as the observed sample, and may eventually not be counted towards the *p*-value. A good computational strategy for comparing probabilities and deciding upon equality of floating point numbers is therefore crucial for a correct implementation of exact test procedures. A permutation test will not resolve the problem of ties, because it also relies on the comparison of the probability of the observed sample with those of other, possibly tied outcomes generated under the null distribution. The exact *p*-values obtained by two different algorithms or on two different computers may not be the same, due to finite precision in the comparison of floating point numbers. Such differences are often explained by ties. If A is the observed table of genotype counts, and table B a tied outcome, for one algorithm the difference in their probability may be less than the tolerance used in the floating point comparison, so that the probability of B is correctly included in the *p*-value. For another algorithm, which potentially calculates the probabilities of the tables with numerical operations that are carried out in a different order, the difference in their probability may exceed the tolerance, such that B is incorrectly not counted towards the final *p*-value. If the probability of table B is large, then the difference in *p*-value due to the ties issue can be large too.

Test results for STRs (see Figure 4b) show more evidence against HWE for multiallelic variants. At first sight, this may suggest multiallelic variants are more prone to genotyping error. This is, however, hard to tell because the statistical power of tests for disequilibrium depends on the distribution of the allele frequencies, and tests have less power at low MAF (Graffelman & Moreno, 2013; Wigginton et al., 2005). On the other hand, if all multiallelic variants are recoded as biallelic (A the most common allele, B any other allele) then the fraction of significant variants remains increasing with the number of alleles, finally indicating that there is apparently more disequilibrium in multiallelic variants.

The comparison of execution times of different algorithms reflects the performance of current state-of-the-art functions in the R environment. The comparison is facilitated by the fact that all execution time measurements were made inside the R environment on the same Linux cluster. However, observed differences are not only due to the algorithm being used, but inevitably also to the coding of the algorithms. It is well known that loops are slower in R than in C, C++ or Fortran, and consequently, many R programs can be speeded up by recoding parts in one of these programming languages. In Figure 3b, on triallelic variants we used an enumeration algorithm, which was fully written in R, whereas large part of the network algorithm was written in C. Therefore, the better execution time of the network algorithm can at least in part be ascribed to its coding in C. If the enumeration algorithm had been coded in C, probably a less striking difference between the two algorithms would have been observed.

In summary, we have made progress in obtaining a great computational improvement for exact HWE testing at three and four allelic X chromosomal variants. The advantage of exact tests is that they do not rely on approximation. It remains a challenge to further improve algorithms and coding for the exact testing of variants with more alleles.

6 | SOFTWARE

The network algorithm for the X chromosomal exact test with multiple alleles is implemented in function HWNetwork of version 1.7.1 of the R-package HardyWeinberg (Graffelman, 2015). We adapted C code for an autosomal network algorithm from Engels' HWxtest package available at https://github.com/wrengels/HWxtest. We imported C functions into R through the Rcpp package (Eddelbuettel & Francois, 2011). An R script reproducing the test results reported in Tables 1 and 2 is available at the Dryad Digital Repository (https://doi.org/10.5061/dryad.8sf7m0cm1).

ACKNOWLEDGEMENTS

This work was supported by grant RTI2018-095518-B-C22 (MCIU/AEI/FEDER) of the Spanish Ministry of Science, Innovation and Universities and the European Regional Development Fund, and by grant R01 GM075091 from the United States National Institutes of Health. We thank three anonymous reviewers whose comments have helped us to improve the article.

AUTHOR CONTRIBUTIONS

J.G. conceived the analysis and the article. L.O. wrote computer programs in C. Both authors performed data analysis and contributed to the writing of the article.

DATA AVAILABILITY STATEMENT

The TSI genotype data are available at www.internationalgenome.org. The example polymorphisms in Table 1 have been included as a data object TSIXTriAllelics in R-package HardyWeinberg. The X chromosomal STR Guizhou Han data used in Section 4.2 are available as supporting information of the article by Chen et al., (2018).

ORCID

Jan Graffelman https://orcid.org/0000-0003-3900-0780

REFERENCES

Aoki, S. (2003). Network algorithm for the exact test of Hardy-Weinberg proportion for multiple alleles. *Biometrical Journal*, *45*(4), 471–490.

Chang, C. C., Chow, C. C., Tellier, L. C. A. M., Vattikuti, S., Purcell, S. M., & Lee, J. J. (2015). Second-generation PLINK: Rising to the challenge of larger and richer datasets. *GigaScience*, *4*(7), 1–16.

Chen, B., Cole, J. W., & Grond-Ginsbach, C. (2017). Departure from Hardy Weinberg equilibrium and genotyping error. *Frontiers in Genetics*, *8*, 167. https://doi.org/10.3389/fgene.2017.00167.

Chen, P., He, G., Zou, X., Wang, M., Jia, F., Bai, H., Li, J., Yu, J., & Han, Y. (2018). Forensic characterization and genetic polymorphisms of 19 X-chromosomal STRs in 1344 Han Chinese individuals and comprehensive population relationship analyses among 20 Chinese groups. *PLoS One*, *13*(9), e0204286. https://doi.org/10.1371/journal.pone.0204286.

Eddelbuettel, D., & Francois, R. (2011). Rcpp: Seamless R and C++ integration. *Journal of Statistical Software*, *40*(8), 1–18.

Engels, W. R. (2009). Exact tests for Hardy-Weinberg proportions. *Genetics*, *183*, 1431–1441.

Fadista, J., Manning, A. K., Florez, J. C., & Groop, L. (2016). The (in)famous GWAS P-value threshold revisited and updated for low-frequency variants. *European Journal of Human Genetics*, *24*(8), 1202–1205. https://doi.org/10.1038/ejhg.2015.269.

Graffelman, J. (2015). Exploring diallelic genetic markers: the HardyWeinberg package. *Journal of Statistical Software*, *64*(3), 1–23.

Graffelman, J. (2020). *Statistical tests for the Hardy-Weinberg equilibrium*. Wiley StatsRef: Statistics Reference Online. https://doi.org/10.1002/9781118445112.stat08274.

Graffelman, J., & Moreno, V. (2013). The mid -value in exact tests for Hardy-Weinberg Equilibrium. *Statistical Applications in Genetics and Molecular Biology*, *12*(4), 433–448.

Graffelman, J., & Weir, B. S. (2016). Testing for Hardy-Weinberg equilibrium at bi-allelic genetic markers on the X chromosome. *Heredity*, *116*(6), 558–568. https://doi.org/10.1038/hdy.2016.20.

Graffelman, J., & Weir, B. S. (2018). Multi-allelic exact tests for Hardy-Weinberg equilibrium that account for gender. *Molecular Ecology Resources*, *18*(3), 461–473. https://doi.org/10.1111/1755-0998.12748.

Graves, J. A. M., Wakefield, M. J., & Toder, R. (1998). The origin and evolution of the pseudoautosomal regions of human sex chromosomes. *Human Molecular Genetics*, *7*(13), 1991–1996.

Guo, W. S., & Thompson, E. A. (1992). Performing the exact test of Hardy-Weinberg proportion for multiple alleles. *Biometrics*, *48*(2), 361–372.

Hosking, L., Lumsden, S., Lewis, K., Yeo, A., Mccarthy, L., Bansal, A., Riley, J., Purvis, I., & Xu, C. (2004). Detection of genotyping errors by Hardy-Weinberg equilibrium testing. *European Journal of Human Genetics*, *12*(5), 395–399.

Huber, M., Chen, Y., Dinwoodie, I., Dobra, A., & Nicholas, M. (2006). Monte Carlo algorithms for Hardy-Weinberg proportions. *Biometrics*, *62*(1), 49–53.

Knaus, B. J., & Grünwald, N. J. (2017). VCFR: a package to manipulate and visualize variant call format data in R. *Molecular Ecology Resources*, *17*, 44–53.

Leal, S. M. (2005). Detection of genotyping errors and pseudo-SNPs via deviations from Hardy-Weinberg equilibrium. *Genetic Epidemiology*, *29*, 204–214.

Levene, H. (1949). On a matching problem arising in genetics. *The Annals of Mathematical Statistics.*, *20*(1), 91–94.

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

Mehta, C. R., & Patel, N. R. (1983). A network algorithm for performing Fisher's Exact Test in r x c contingency tables. *Journal of the American Statistical Association*, *78*(382), 427–434.

Mills, R. E., Luttig, C. T., Larkins, C. E., Beauchamp, A., Tsui, C., Pittard, W. S., & Devine, S. E. (2006). An initial map of insertion and deletion (INDEL) variation in the human genome. *Genome Research.*, *16*(9), 1182–1190. https://doi.org/10.1101/gr.4565806.

Panagiotou, O. A., Ioannidis, J. P., The Genome-Wide Significance Project (2012). What should the genome-wide significance threshold be? Empirical replication of borderline genetic associations. *International Journal of Epidemiology*, *41*(1), 273–286. https://doi.org/10.1093/ije/dyr178.

Purcell, S., Neale, B., Todd-Brown, K., Thomas, L., Ferreira, M. A. R., Bender, D., Maller, J., Sklar, P., de Bakker, P. I. W., Daly, M. J., & Sham, P. C. (2007). PLINK: A toolset for whole-genome association and population-based linkage analysis. *American Journal of Human Genetics*, *81*(3), 559–575.

R Core Team (2020). *R: A language and environment for statistical computing*. Vienna, Austria: R Foundation for Statistical Computing. http://www.R-project.org/.

Roeder, K., & Wasserman, L. (2009). Genome-wide significance levels and weighted hypothesis testing. *Statistical Science*, *24*(4), 398–413. https://doi.org/10.1214/09-STS289.

Teo, Y. Y., Fry, A. E., Clark, T. G., Tai, E. S., & Seielstad, M. (2007). On the usage of HWE for identifying genotyping errors. *Annals of Human Genetics*, *71*(5), 701–703.

The 1000 Genomes Project Consortium (2015). A global reference for human genetic variation. *Nature*, *526*, 68–74.

Wigginton, J. E., Cutler, D. J., & Abecasis, G. R. (2005). A note on exact tests of Hardy-Weinberg equilibrium. *The American Journal of Human Genetics*, *76*, 887–893.

Xu, C., Tachmazidou, I., Walter, K., Ciampi, A., Zeggini, E., Greenwood, C. M. & the UK10K Consortium (2014) Estimating genome-wide significance for whole-genome sequencing studies. *Genetic Epidemiology*, *38*(4), 281–290. https://doi.org/10.1002/gepi.21797.

Ziegler, A., & König, I. R. (2006). *A statistical approach to genetic epidemiology*. Weinheim: Wiley-VCH Verlag.

How to cite this article: Graffelman J, Ortoleva L. A network algorithm for the X chromosomal exact test for Hardy–Weinberg equilibrium with multiple alleles. *Mol Ecol Resour*. 2021;21:1547–1557. https://doi.org/10.1111/1755-0998.13373