Genetic Epidemiology 30: 320–332 (2006)

An Evaluation of the Replicate Pool Method: Quick Estimation of Genome-Wide Linkage Peak *p*-Values

Janis E. Wigginton* and Gonçalo R. Abecasis

Center for Statistical Genetics, Department of Biostatistics, University of Michigan, Ann Arbor, MI

The calculation of empirical *p*-values for genome-wide non-parametric linkage tests continues to present significant computational challenges for many complex disease mapping studies. The gold standard approach is to use gene dropping to simulate null genome scans. Unfortunately, this approach is too computationally expensive for many data sets of interest. An alternative, more efficient method for sampling null genome scans is to pre-calculate pools of family-specific statistics and then resample from these replicate pools to generate ''pseudo-replicate'' genome scans. In this study, we use simulations to explore properties of the replicate pool *p*-value estimator $\hat{p}_{RP}$ and show that it provides an excellent approximation to the traditional gene-dropping estimator for significantly less computational effort. While the computational efficiency of the replicate pool estimator is noticeable in almost all data sets, by applying the replicate pool method to several previously characterized data sets we show that savings in computational effort can be especially significant (on the order of 10,000-fold or more) when one or more large families are analyzed. We also estimate replicate pool *p*-values for the schizophrenia data described by Abecasis et al. and show that $\hat{p}_{RP}$ closely approximates gene-drop *p*-values for all linkage peaks reported for this study. Lastly, we expand upon Song et al.'s previous work by deriving a conservative estimator of the variance for $\hat{p}_{RP}$ that can easily be computed in practical settings. We have implemented the replicate pool method along with our variance estimator in a new program called Pseudo, which is the first widely available automated implementation of the replicate pool method. *Genet. Epidemiol.* 30:320–332, 2006. © 2006 Wiley-Liss, Inc.

Key words: non-parametric linkage analysis; significance levels; Monte-Carlo methods

Contract grant sponsor: National Human Genome Research Institute; Contract grant sponsor: National Eye Institute.
*Correspondence to: J. E. Wigginton, Department of Biostatistics, School of Public Health, University of Michigan, Ann Arbor, MI.
E-mail: wiggie@umich.edu
Received 5 December 2005; Revised 14 February 2006; Accepted 14 February 2006
Published online 3 April 2006 in Wiley InterScience (www.interscience.wiley.com).
DOI: 10.1002/gepi.20147

INTRODUCTION

Evaluating the statistical significance of genome scan linkage peaks is important, not only to prioritize regions for follow-up studies, but also to critically evaluate the success of the gene-mapping experiment [Ott, 1989]—if the results are compatible with the null, then alternative gene-mapping approaches should be considered or additional samples should be collected to ensure adequate power.

The problem is deceptively simple: given an observed genome-wide peak for a linkage statistic, we would like to determine the probability of observing a greater or equal peak statistic by chance. While fairly straightforward in principle, this problem is quite challenging because the null distribution of peaks can be difficult to characterize for most linkage statistics. This null distribution depends on the type of families being examined, the distribution of affected and unaffected individuals within these families, the degree of polymorphism and spacing of markers available for analysis, the pattern of missing genotype data, and—naturally—the statistic of interest [Ott, 1989]. In practice, even approximate representations of the null distribution, constructed using either analytical or empirical methods have proven very useful [Teng and Siegmund, 1997; Kruglyak et al., 1998; Bacanu et al., 2005].

One of the earliest interpretations for a genome scan linkage peak was presented by Newton Morton [1955], in the context of LOD score analysis. Morton used the prior probability of linkage to reason that for a simple Mendelian trait a LOD score of 3.0 would yield a false positive rate of 5%. Since then, the focus of gene mapping

has shifted from simple Mendelian traits to more complex traits, and gene mapping strategies—both in vitro and in silico—have undergone many changes. Nevertheless, Morton's argument is simple, and the result remains relatively accurate in many settings [Lander and Kruglyak, 1995; Morton, 1998; Elston, 1998].

Nowadays, most linkage studies rely on non-parametric statistics, such as the NPL score [Whittemore and Halpern, 1994; Kruglyak et al., 1996] or the Z_{lr} statistic [Kong and Cox, 1997]. It is well known that appropriate significance thresholds for these statistics depend on features of each specific data set, such as marker informativeness and spacing as well as the family structures being examined [Kruglyak et al., 1996; Sawcer et al., 1997; Kong and Cox, 1997; Kruglyak and Daly, 1998].

Ott [1989] advocated the use of Monte-Carlo approaches for evaluating the significance of observed linkage peaks. When carefully designed and implemented, Monte-Carlo methods are computationally intensive but can more accurately reflect the ''quirks'' of each data set and analysis strategy. In their simplest form, these Monte-Carlo methods use hundreds or thousands of gene-dropping simulations to generate replicate genomes which can then be used to reproduce the null distribution of any statistic of interest. In the context of linkage analysis, calculation of the linkage statistic on each of these replicate genomes can still require substantial amounts of computing power for many interesting data sets.

In recent years, the development of efficient Monte-Carlo algorithms for evaluating significance levels has become an active area of research. One area of focus has been the development of algorithms that can estimate accurate significance levels using small numbers of replicates. Several promising approaches include the use of sequential stopping rules that analyze fewer replicates to evaluate less significant findings [Besag, 1991], and a replicate pool method, originally suggested by Terwilliger and Ott [1992] and revisited in detail by Song et al. [2004]. Because the replicate pool method has the virtue of being unbiased, extremely efficient and readily implemented using output from existing linkage packages, we believe that it is an extremely attractive option.

In this article, we describe and evaluate the accuracy and computational efficiency of algorithms for evaluating the significance of single or multiple genome scan linkage peaks using the replicate pool method. The method can be applied to multiple correlated traits, and we describe and evaluate a practical strategy for quantifying the accuracy of estimated significance levels by extending the work of Song et al. [2004]. Our methods are implemented in freely available computer code and, to our knowledge, are the first widely available, automated implementation of the replicate pool method for estimating genome scan significance levels.

METHODS

BACKGROUND

Data for a genome scan will generally include genotype and phenotype information for a set of related individuals. Phenotypes P may include measurements for one or more traits and genotypes G will usually include a set of markers distributed throughout the genome along a marker map M. The goal is to determine if, within each family, phenotypically similar individuals share particular chromosomal regions more frequently than expected by chance. A typical test of linkage assesses evidence for a genetic effect within a collection of n families by constructing a combined linkage statistic (CLS)

$$\mathrm{CLS} = h(Z_1, Z_2, \ldots, Z_n)$$

where family-specific scores $Z_1, Z_2, \ldots, Z_n$ quantify evidence for genetic sharing within each family $1 \ldots n$.

If P_f and G_f represent, respectively, the observed phenotype and genotype data for family f, a score for the family may be defined as

$$Z_f = g(P_f, G_f, M, R_f)$$

where g measures genetic sharing within the observed family data relative to what would be expected in the absence of any genetic effect, given family relationships R_f and marker map M.

Since the true null distribution of the CLS for a specific data set may be unknown, significance is often assessed empirically. An empirical null distribution can be generated by repeating the original linkage scan in simulated data sets that reproduce the patterns of marker informativeness, spacing and missing genotypes in the original data [Terwilliger and Ott, 1994].

REPLICATE POOL APPROACH

Since generation of the empirical null distribution for h calls for recalculation of the CLS profile thousands of times, any algorithmic improvement

that targets critical portions of this calculation can significantly impact overall computation time. For the CLS statistic, the calculation of family-specific scores tends to be the rate-limiting step primarily because ambiguities in inheritance patterns due to non-informative markers or missing data make it necessary to integrate over many alternatives compatible with the observed data. The replicate pool method (illustrated in Fig. 1) specifically targets this bottleneck by performing the costly calculation for only a small set of replicates. Once initial pools of family-specific scores have been pre-calculated, additional "pseudosimulations" are generated by independently sampling one score for each family from its own pool of pre-calculated statistics and recalculating the overall score on the selected sample:

$$\mathrm{CLS}_{RP} = h(Z_{1j}, Z_{2k}, \ldots, Z_{nl}).$$

Here, Z_{fp} represents the pth replicate for family f in the pool of pre-calculated family-specific scores.

APPLICATION OF THE REPLICATE POOL METHOD TO KONG AND COX TESTS

In recent years, non-parametric tests using the Kong and Cox linear model [Kong and Cox, 1997] have gained general acceptance and are widely used in complex disease mapping studies. The idea is to use a scoring function S_f [Whittemore and Halpern, 1994; McPeek, 1999] that ranks each inheritance vector according to the evidence for linkage it provides. Given a specific choice of scoring function, evidence for linkage within a family is modeled by a normalized expected score:

$$Z_f = (E(S_f|G) - \mu_f)/\sigma_f$$

where μ_f and σ_f represent the expected mean and standard deviation of the statistic S_f under the assumption of no linkage. Calculating $E(S_f|G)$ involves iterating over possible inheritance vectors and the number of inheritance vectors to be considered rises exponentially with family size. In some data sets, this step can require several hours of processor time to analyze. When there is no uncertainty about inheritance, the scores Z_f are normally distributed with mean 0 and variance 1, and overall evidence for linkage can be evaluated by simply taking a weighted sum of the individual statistics [Kruglyak et al., 1996]. The approach of Kong and Cox [1997] was specifically developed to deal with settings where there is uncertainty about inheritance because the available genetic markers are not fully informative. Overall evidence for linkage is modeled by finding the value of an allele sharing parameter δ_{hat} that maximizes

$$\mathrm{LOD}_{KC} = \Sigma_f \log_{10}(1 + \delta_{\mathrm{hat}} Z_f).$$

At a single point in the genome, the resulting LOD score statistic is approximately distributed as a

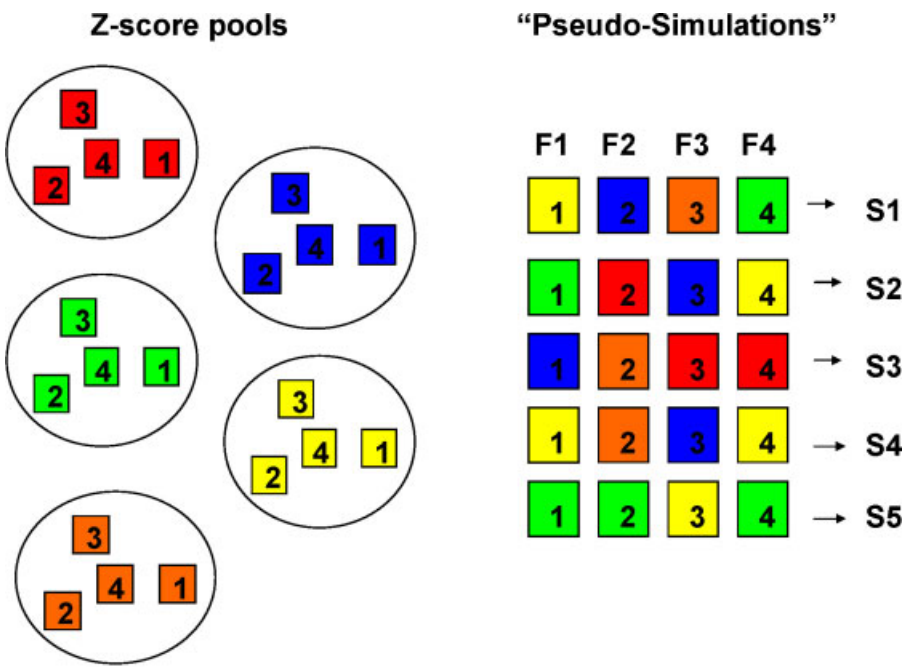

Fig. 1. Replicate pool method. Starting with a pool consisting of a small number of pre-calculated family-specific scores, "pseudo-simulations" are generated by randomly sampling one *z*-score per family and recalculating the overall linkage score.

50:50 mixture of $\chi^2/2\log(10)$ and a point mass at zero. However, the distribution of genome-wide peak statistics is less clear since the dependence between neighboring locations is influenced both by family structure and marker informativeness.

APPLICATION TO GENOME-WIDE LINKAGE STATISTICS

Although we have considered only a single analysis position up to this point in the discussion, in most settings, multiple positions located throughout the genome will be analyzed. The observed result will be a profile of linkage scores

$$\text{PROF} = (\text{CLS}_1, \text{CLS}_2, \text{CLS}_3, \ldots, \text{CLS}_p)$$

located at p positions along the genome. This profile will include a linkage peak whenever the statistic reaches a local maximum. We denote the highest of these local maxima as M_1, the second highest as M_2, and so on, to define a set of peaks $(M_1, M_2, \ldots, M_k)$. To properly evaluate the significance of these linkage peaks, it is crucial to account for the correlation between linkage statistics at neighboring positions. The replicate pool method is especially attractive in this setting since the computational cost of gene-dropping simulations for whole genome scans rapidly becomes prohibitive.

To extend the basic approach to multiple positions, instead of sampling individual statistics we sample blocks B_{cf} of z-scores for family f for all positions along chromosome c. Pseudo-replicate genome scans are constructed by sampling one block of z-scores for each family and chromosome and then calculating Kong and Cox LODs for all p analysis positions. Each "pseudo-simulation" now replicates a genome scan drawn from the null distribution of PROF and results in a profile of simulated LOD scores.

ESTIMATION OF SIGNIFICANCE

Once the empirical distribution for the lod score profile (PROF) has been constructed (either by gene-dropping or the replicate pool method), empirical p-values for observed peaks can be assigned by ranking all maxima of interest in descending order $(M_1, M_2, \ldots, M_k)$. A p-value for the highest score (M_1) is calculated as

$$p(M_1) = N(\max(\text{CLS}) > M_1)/N \approx P(\max(\text{CLS}) > \text{M}_1)$$

where $N(\max(\text{CLS}) > M_1)$ is the number of simulated genome scans where the maximum lod score (max(CLS)) is greater than M_1.

As an alternative to considering only the highest peak, we can consider a group of peaks jointly [Wiltshire et al., 2002]. For instance, to evaluate whether more peaks exceed M_j than expected by chance, we calculate

$$\begin{aligned} p(M_1, M_2, \ldots, M_j) &= N(\max_j(\text{CLS}) > M_j)/N \\ &\approx P(\max_j(\text{CLS}) > M_j) \end{aligned}$$

where $\max_j(\text{CLS})$ is the jth highest independent linkage score for a simulated genome. If this probability is small, we reject the null hypothesis that all peaks occurred by chance and accept the alternative hypothesis that at least one of the peaks is real. This approach can provide for a more powerful linkage test when there are multiple loci of modest effect, rather than a single large locus. For simplicity, we treat linkage peaks as independent if and only if they occur on different chromosomes; however, our approach can be extended to more flexible settings.

VARIANCE FOR THE REPLICATE POOL METHOD

An important feature of the replicate pool method is that the variance structure for the replicate pool estimator ($\hat{p}_{\text{RP}}$) differs from the gene-dropping estimator ($\hat{p}_{\text{GD}}$). For N simulations, the distribution of $N*\hat{p}_{\text{GD}}$ will be Binomial(N, p) and

$$\text{Var}(\hat{p}_{\text{GD}}) = p(1-p)/N.$$

The variance of $\hat{p}_{\text{RP}}$ is harder to estimate because family-specific statistics are reused, inducing a correlation among simulated outcomes. Song et al. [2004] studied the variance structure of $\hat{p}_{\text{RP}}$. They considered settings where N_{GD} gene dropping simulations are done to estimate $\hat{p}_{\text{GD}}$, and then family-specific outcomes of these gene-dropping simulations are resampled to generate N_{RP} pseudo-replicates and estimate $\hat{p}_{\text{RP}}$. They demonstrated that when $N_{\text{RP}} \gg N_{\text{GD}}$, the pseudo-replicate-based p-value is more efficient. They also showed that

$$\begin{aligned} \text{Var}(\hat{p}_{\text{RP}}) &\approx p/N_{\text{RP}} + p(\Sigma_f \Sigma_c W_{fc} - N_F * N_C * p)/N_{\text{GD}} \\ &\approx p(\Sigma_f \Sigma_c W_{fc} - N_F * N_C * p)/N_{\text{GD}} \\ &\quad (\text{when } N_{\text{RP}} \text{ is large}) \end{aligned} \tag{1}$$

where W_{fc} is the variance weight for family f on chromosome c, N_F is the number of families and N_C is the number of chromosomes. Their results

lead to rough bounds on $\mathrm{Var}(\hat{p}_{\mathrm{RP}})$ of

$$p(1-p)/N_{\mathrm{RP}} < \mathrm{Var}(\hat{p}_{\mathrm{RP}}) < p(1-p)/N_{\mathrm{GD}}$$

since the summation in equation (1) must be strictly positive and $p\ (1–p)/N_{\mathrm{RP}} < p/N_{\mathrm{RP}}$.

FAMILY VARIANCE WEIGHTS

Family variance weights (W_{fc}) vary between 0.0 and 1.0 and represent the impact of each sampling unit on the final linkage statistic. In our case, the sampling units consist of a chromosome/family pair combination. Intuitively, we expect that data for larger pedigrees and longer chromosomes might have a greater impact in the overall linkage profile. Formally, Song et al. [2004] defined W_{fc} by considering two abstract samples from the replicate pools, say A and A*, that include the same data for sampling unit *fc* but differ in all other respects, say $\mathrm{A} \cap \mathrm{A}^* = \{A_{fc}\}$. For a specific family *f* and chromosome *c*, they then defined $W_{fc} = P(\mathrm{PROF(A}^*) \text{ rejects } H_0 \mid \mathrm{PROF(A)} \text{ rejects } H_0)$ when $\mathrm{A} \cap \mathrm{A}^* = \{A_{fc}\}$.

Thus, W_{fc} will be large and close to 1.0 if sampling unit *fc* alone can shape the linkage profile. Conversely, if sampling unit *fc* has little impact on the linkage profile, W_{fc} will be small and close to the unconditional probability that PROB(A*) rejects H_0.

ESTIMATION OF FAMILY VARIANCE WEIGHTS

Exact analytical calculation of these variance weights is difficult because our test statistic is not a simple sum of the individual test statistics. However, approximate confidence bounds can be calculated using a conservative estimator that relies on information that can be easily gathered while generating pseudo-replicate samples. After testing a number of candidates, we found that a reasonably accurate but conservative estimator for the variance weight is

$$\hat{W}_{fc} = \Sigma_r (p_{rfc}) n^+_{rfc} / N^+$$

where

$$p_{rfc} = \begin{cases} n^+_{rfc}/n_{rfc} & \text{when } n_{rfc} > 0 \\ 0 & \text{when } n_{rfc} = 0. \end{cases}$$

In these formulae, N^+ is the total number of pseudo-replicate genome scans that reject H_0, *r* indexes the blocks of statistics in each replicate pool, n^+_{rfc} is the number of pseudo-replicate genome scans that reject H_0 and include block *r* from pool *fc and* n_{rfc} is the number of pseudo-replicate genome scans that include block *r* from pool *fc* (whether or not they reject H_0).

SIMULATION STUDIES

We performed simulations to evaluate the replicate pool method and compare it to gene dropping. For all simulations, replicate pool *p*-values were calculated using our implementation of the replicate pool method (Pseudo, freely available with C++ source code at http://www.sph.umich.edu/csg/abecasis/pseudo/). Gene-dropping *p*-values were estimated using replicates generated by Merlin's [Abecasis et al., 2002; Abecasis and Wigginton, 2005] simulation engine and analyzed by Merlin's implementation of the Kong and Cox linear model with S_{all} as the scoring function. Replicate pool *p*-values were estimated by generating pseudo-replicate genome scans, each constructed by sampling linkage statistics one chromosome and family at a time. All replicate pool *p*-values were estimated solely from simulated pseudo-replicate genome scans, excluding the initial null replicates. Each run of the replicate pool method was based on a newly simulated pool of *z*-scores.

Comparision of *p*-values. To compare the distribution of $\hat{p}_{\mathrm{RP}}$ and $\hat{p}_{\mathrm{GD}}$, we simulated samples with 100 nuclear families in four different configurations: (1) two affected siblings and both parents genotyped, (2) three affected siblings and both parents genotyped, (3) three affected siblings and neither parent genotyped and (4) a mixed sample with untyped parents containing 55 families with two affected siblings, 35 families with three affected siblings and five families with four affected siblings. For each sample, $\hat{p}_{\mathrm{GD}}$ was estimated once using 10,000 gene-dropping simulations. This estimate was compared to 50 independent point estimates of p_{RP} each obtained by generating 50,000 pseudo-replicate genome scans by sampling from a pool of 50 initial gene-dropping analyses. This comparison was performed for each of three different maps: (1) a 1 cM microsatellite map (2) a 10 cM microsatellite map and (3) a 1 cM SNP map.

For each map type, markers were evenly spaced along 22 autosomes, with lengths of 289, 270, 235, 216, 213, 189, 198, 172, 179, 180, 163, 175, 137, 127, 130, 127, 142, 130, 116, 108, 75 and 72 cM for chromosomes 1–22, respectively. Markers for microsatellite maps were simulated with a minimum of seven and a maximum of 10 alleles, with the number of alleles for each marker chosen randomly from this range. Average heterozygosity

for microsatellites was approximately 84%. All SNP maps were simulated with two alleles per marker and average heterozygosity of 45%. For each map type, allele frequencies were simulated by randomly selecting a number within a restricted range and then normalizing all values by their sum. This guaranteed that simulated allele frequencies fell in reasonable ranges compared to those we have observed in experimentally derived data and that overall frequencies summed to 1.0.

Applied example. We also compared *p*-values estimated by the replicate pool method to reported gene-drop *p*-values for a previously characterized data set [Abecasis et al., 2004a]. The data consist of 143 Afrikaner schizophrenia families ranging in size from 3 to 16 individuals and including 137 affected relative pairs (including 79 parent-child pairs, 35 sib pairs, 15 avuncular pairs, 4 grandparent-grandchild pairs, and 4 cousin pairs). Evaluated phenotypes included three nested phenotypic liability classes consisting of core schizophrenic phenotypes (LCI), individuals diagnosed with a psychotic disorder (LCII) and individuals diagnosed with any psychiatric disorder (LCIII). Thirty-four of 143 families are informative for linkage and include 62, 81, and 100 individuals in LCs I, II, and III, respectively. Remaining individuals are used for allele frequency estimation. Genotypes were collected for 388 microsatellites composed primarily of trinucleotide and tetranucleotide repeats with average spacing of 9 cM (including one gap of >20 cM) and average heterozygosity of .76. The entire genotyped sample included 173, 205, and 253 individuals in LCs I, II, and III, respectively.

Significance for all LOD score peaks >1.0 was determined in the original study using 5000 gene-dropping simulations. We compare the most significant results in the original study with those obtained using an initial pool of 50 *z*-score replicates per family to generate 50,000 pseudo-replicate genome scans.

Computational efficiency. To quantify computing resources we ran 20 gene-dropping simulations for each of three data sets. We then resampled the resulting statistics to generate 100,000 pseudo-replicate genome-scans. We recorded the total running time, average running time per simulation and memory use for each stage of the analysis. Running times for $N = 1000$, 10,000 and 100,000 gene-dropping simulations were extrapolated from these simulations.

Relative efficiency of replicate pool and gene-dropping methods. We assessed relative efficiency of the two methods by comparing sample variances of $\hat{p}_{RP}$ and $\hat{p}_{GD}$ in samples containing 16–128 nuclear families. For each sample, we generated either 20 or 50 gene-dropping replicates and performed linkage analysis on each replicate. Linkage results were used to calculate $\hat{p}_{GD}$ and family-specific *z*-scores were saved and used to seed subsequent runs of the replicate pool method, each generating 100,0000 pseudo-replicates to estimate $\hat{p}_{RP}$. This analysis was repeated 50 times (with each run using a new set of seed replicates) allowing us to evaluate the empirical variance of $\hat{p}_{RP}$ and $\hat{p}_{GD}$ for each combination of family size, number of initial replicates and LOD score threshold. Finally, relative efficiency was defined as $RE = Var(p_{GD})/Var(p_{RP})$.

Development of family weight estimator. Although direct comparison of estimates to true variances is obviously the better gauge of performance for a family weight estimator, exhaustive testing was computationally prohibitive for all candidate estimators. Instead, we constructed a test scenario (detailed in supplemental information) where true weights could be calculated directly from the data. By comparing true weights to those predicted by each candidate estimator under a wide variety of sampling conditions, we were able to refine our selection to several reasonable estimators that were conservative under almost all sampling conditions.

Variance estimation for $\hat{p}_{RP}$ in genome scan data. Our final selection of a family weight estimator $\hat{W}_{fc}$ was tested by directly comparing estimated variances for $\hat{p}_{RP}$ to the empirical variance of $\hat{p}_{RP}$ estimated from simulation. The behavior of STD_{RP} (the estimated standard deviation of $\hat{p}_{RP}$ with family weight probabilities estimated by $\hat{W}_{fc}$), was examined for various initial pool sizes, number of families and pseudo-replicates.

RESULTS

COMPARISON OF *p*-VALUES

We first checked whether results obtained using $\hat{p}_{RP}$ accurately reproduce those obtained with $\hat{p}_{GD}$. Our simulations suggest that the *p*-values estimated using $\hat{p}_{RP}$ (the replicate pool statistic) provide an extremely good approximation to the *p*-values estimated using $\hat{p}_{GD}$ (the more computationally intensive gene-dropping statistic). Table I shows estimated significance levels for various LOD score thresholds estimated using either $\hat{p}_{RP}$ or $\hat{p}_{GD}$ for 1 cM microsatellite, 1 cM SNP, or 10 cM

TABLE I. Comparison of *p*-values estimated using either gene-dropping or replicate pools

		Gene-dropping *p*-value ($N_{GD} = 10{,}000$)		Replicate pool *p*-value (50 trials each with $N_{GD} = 50$, $N_{RP} = 50{,}000$)		
Family configuration	LOD threshold	p_{GD}	SE(p_{GD})	Estimates from first five trials	Mean estimate	SD
1 cM map, 7–10 alleles per marker						
. . .2 children+parents genotyped	3.0	.1132	.00317	(.10220, .11126, .11448, .11792, .11220)	.11378	.005833
	3.5	.0382	.00192	(.03408, .03854, .03944, .04003, .03862)	.03852	.002369
	4.0	.0115	.00107	(.01184, .01292, .01334, .01320, .01234)	.01292	.009300
. . .3 children+parents genotyped	3.0	.1035	.00305	(.10646, .10798, .08958, .09030, .09802)	.09976	.005379
	3.5	.0374	.00190	(.03598, .03690, .02906, .02972, .03448)	.03381	.002238
	4.0	.0122	.00110	(.01176, .01206, .00938, .01008, .01076)	.01114	.000804
. . .3 children+parents not genotyped	3.0	.0876	.00283	(.09158, .09672, .09490, .08870, .08492)	.08727	.004744
	3.5	.0316	.00175	(.03090, .03264, .03328, .03022, .02798)	.02948	.002035
	4.0	.0110	.00104	(.00950, .01164, .01032, .00984, .00906)	.00963	.000794
. . .Mixed family sample	3.0	.0926	.00290	(.09244, .09016, .08922, .09524, .09310)	.09453	.004971
	3.5	.0312	.00174	(.03242, .03042, .03076, .03188, .03194)	.03242	.002105
	4.0	.0100	.00099	(.01062, .00980, .01012, .01042, .01184)	.01088	.000974
1 cM SNP map, 7–10 alleles per marker						
. . .2 children+parents genotyped	3.0	.0891	.00285	(.09260, .08202, .03324, .08088, .08534)	.09029	.005701
	3.5	.0307	.00173	(.03118, .02714, .03074, .02734, .03016)	.03088	.002239
	4.0	.0103	.00101	(.01004, .00882, .01064, .00906, .00928)	.01021	.000887
. . .3 children+parents genotyped	3.0	.0775	.00267	(.08594, .08358, .07663, .08542, .08144)	.08239	.003898
	3.5	.0240	.00153	(.02930, .02832, .02594, .02920, .02746)	.02767	.001620
	4.0	.0081	.00090	(.01074, .00950, .00792, .00924, .00906)	.00968	.000791
. . .3 children+parents not genotyped	3.0	.0602	.00238	(.05470, .05836, .05314, .05572, .06004)	.05757	.003209
	3.5	.0220	.00147	(.01888, .01922, .01758, .01714, .01964)	.01896	.001383
	4.0	.0064	.00080	(.00596, .00668, .00588, .00588, .00662)	.00612	.000571
. . .Mixed family sample	3.0	.0607	.00239	(.06394, .05620, .06058, .05568, .06588)	.06056	.004876
	3.5	.0191	.00137	(.02000, .01958, .02066, .01798, .02180)	.02019	.001744
	4.0	.0051	.00071	(.00658, .00660, .00682, .00558, .00692)	.00660	.000653
10 cM map 7-10 alleles per marker						
. . .2 children+parents genotyped	3.0	.0446	.00206	(.04124, .05060, .04044, .04298, .04712)	.04461	.003267
	3.5	.0157	.00124	(.01300, .01660, .01426, .01398, .01576)	.01468	.001138
	4.0	.0045	.00067	(.00414, .00508, .00420, .00508, .00476)	.00472	.000437
. . .3 children+parents genotyped	3.0	.0378	.00191	(.03808, .03872, .03758, .04348, .04098)	.03973	.002809
	3.5	.0125	.00111	(.01238, .01158, .01184, .01402, .01314)	.01278	.001081
	4.0	.0031	.00055	(.00374, .00366, .00392, .00490, .00338)	.00411	.000424
. . .3 children+parents not genotyped	3.0	.0353	.00184	(.03942, .03206, .03350, .03542, .03238)	.03397	.002331
	3.5	.0098	.00099	(.01244, .01014, .01054, .01146, .00990)	.01091	.008380
	4.0	.0037	.00061	(.00432, .00324, .00436, .00384, .00314)	.00348	.003720
. . .Mixed family sample	3.0	.0387	.00193	(.03838, .04000, .03820, .03492, .03348)	.03709	.002956
	3.5	.0134	.00115	(.01280, .01402, .01286, .01124, .01112)	.01216	.001206
	4.0	.0032	.00056	(.00444, .00468, .00410, .00360, .00334)	.00401	.000498

Entries in the column labeled ''Estimates from trials 1–5'' are *p*-values estimated by the first five of 50 trials of the replicate pool method. All 50 trials were used to estimate the overall mean and standard deviation for the statistic.

microsatellite maps. In all cases, analysis was carried out using a 1 cM grid. In only one instance did the average of the replicate pool *p*-values fall outside the 99% confidence interval for the gene drop *p*-value. This was the *p*-value for the probability of obtaining a LOD > 3.5 using a 1 cM SNP map in families including three genotyped offspring and no parental genotypes (where the average of $\hat{p}_{RP} = .01896$ but $\hat{p}_{GD} = .0220$). Subsequent simulation work gave gene-drop estimates with an average of $p = .0201$, indicating that the original estimate of the gene-dropping *p*-value fell on the extreme end of the true confidence interval.

APPLIED EXAMPLE USING RANKED *p*-VALUES

As an additional test, we compared *p*-values predicted by the replicate pool method to reported significance levels for an entire set of linkage peaks from a published study [Abecasis et al., 2004a]. As before, the distribution of $\hat{p}_{RP}$ was

TABLE II. Comparison of gene-dropping and replicate pool *p*-values for schizophrenia data set

	Gene-dropping *p*-value ($N_{GD} = 5000$)		Replicate pool *p*-values (50 trials each with $N_{GD} = 50$, $N_{RP} = 50{,}000$)		
Outcome	p_{GD}	SE(p_{GD})	Estimates from trials 1–5	Mean	SD
LCI					
3+peaks with LOD>1.46	.230	.00595	(.23218, .24388, .23704, .24866, .21750)	.24257	.015560
2+peaks with LOD>1.98	.060	.00336	(.02878, .03108, .03134, .03312, .03068)	.03099	.002523
Overall peak LOD>2.28	.072	.00366	(.07244, .07632, .07292, .07792, .06972)	.07390	.003839
LCII					
3+peaks with LOD>1.04	.887	.00448	(.89236, .89600, .89398, .89908, .87342)	.89552	.010430
2+peaks with LOD>1.84	.171	.00532	(.17228, .17782, .17704, .18604, .15950)	.17907	.010335
Overall peak LOD>3.21	.006	.00109	(.00638, .00776, .00800, .00690, .00724)	.00751	.000607
LCII					
3+peaks with LOD>1.12	.771	.00594	(.77566, .78574, .78124, .79380, .76776)	.78502	.015703
2+peaks with LOD>2.20	.035	.00260	(.03508, .03762, .03720, .03896, .03450)	.03667	.002829
Overall peak LOD>3.30	.007	.00118	(.00886, .00962, .00946, .00864, .00946)	.00923	.000709

Entries in the column labeled ''Estimates from trials 1–5'' are *p*-values estimated by the first five of 50 trials of the replicate pool method. All 50 trials were used to estimate the overall mean and standard deviation for the statistic.

represented by 50 trials of the replicate pool method, each using 50 initial replicates to generate 50,000 pseudo-replicate genome scans. Table II compares this distribution to reported values of $\hat{p}_{GD}$ calculated from 5,000 null replicates for each of three liability classes. For two outcomes (the probability of obtaining 3 peaks with LOD>1.12 for LCIII and the probability of obtaining three peaks with LOD>1.46 for LCI) the average $\hat{p}_{RP}$ fell outside of the 95% confidence interval of $\hat{p}_{GD}$. For another outcome (probability of obtaining two or more peaks with LOD>1.98 for LCI), the average $\hat{p}_{RP}$ fell outside the 99% confidence interval for $\hat{p}_{GD}$, and we repeated the original gene-drop simulation three times to obtain a better sense of the true distribution of $\hat{p}_{GD}$. These simulations produced *p*-value estimates of .04006, .03119 and .02879, which suggests that the original published estimate was extreme and that the average predicted by $\hat{p}_{RP}$ (.03099) actually fell well within reasonable confidence bounds for $\hat{p}_{GD}$.

COMPUTATIONAL EFFICIENCY FOR THE REPLICATE POOL METHOD

The replicate pool method is an adaptation of the gene-dropping method that replaces a computationally intensive step (the calculation of *z*-scores for individual families) with a much less demanding procedure (simple resampling of precalculated *z*-scores). Logically, it is to be expected that the replicate pool method will always be more computationally efficient than gene-dropping. While our simulation work confirms this intuition, we also find that the family structure and proportion of missing data for a data set, along with characteristics of the marker map, strongly influence relative computational efficiency of the two methods.

Table III shows processor times required to implement each method in three data sets with differing family size distribution and marker map characteristics. A striking feature of these results is the marked computational savings that are possible when the replicate pool method is used to analyze data sets containing large families. For the data sample described in Abecasis et al. [2004b] which includes four large families with 30, 26, 24 and 23 individuals respectively, we were able to generate 20 seed replicates and perform 100,000 pseudo-simulations in approximately 8.3 days total CPU time. In contrast, completion of the same number of gene-dropping simulations would have required more than a century of processor time. In addition, note that although each seed replicate required up to 2.6 GB of memory to analyze (due to requirements of using the Lander–Green algorithm to represent inheritance vector space), generating and processing 100,000 pseudo-replicate genome scans required <50 Mb of RAM (most of it used to generate lookup tables of the results of the initial gene-dropping simulations). When families are large or factors that make inheritance patterns ambiguous (such as missing data or uninformative markers) are present, we expect the replicate pool

TABLE III. Comparison of processor time and memory requirements for the replicate pool and gene-dropping methods

	Abecasis, 2004a	Abecasis, 2004b	Faraone, 2005
Data set characteristics			
Number of families	143	126	60
Average Family Size	3.36	7.56	7.36
Maximum Family Size	16	30	21
Analysis Positions	370	728	6689
Processor time required for replicate pool method			
Generate 20 seed replicates	10 m 30 s	7 d 8 h	15 d 11 h
Process 1,000 pseudo-simulations	5 m 30 s	12 m 55s	1 h 2 m
Process 10,000 pseudo-simulations	33 m 17 s	2 h 57 m	10 h 2 m
Process 100,000 pseudo-simulations	5 h 6 m	22 h 42 m	4 d 4 h
Estimated processor times for gene dropping method			
Execute 1,000 gene dropping simulations	8 h 50 m	366 d	773 d
Execute 10,000 gene dropping simulations	88 h 12 m	10 yr	21 yr
Execute 100,000 gene dropping simulations	36 d 18 h	>100 yr	>220 yr
Memory requirements			
To process 100,000 pseudo-simulations	9.9 M	25.7 M	45.7 M
For each gene-dropping simulation	9 M	2.6 G	1.9 G

method will be significantly more efficient than gene-dropping.

Once family z-scores have been calculated (in the case of gene-dropping) or selected from a pool of pre-calculated scores (in the case of replicate pool method), the two methods are algorithmically equivalent. In each case, the maximum likelihood estimate of delta is determined and the linkage score is calculated at each position. Thus, factors that impact the efficiency of standard linkage analysis at this stage of the algorithm (i.e. number of analysis positions and number of families) have a similar influence on computation time of both methods.

RELATIVE EFFICIENCY OF REPLICATE POOL METHOD AND GENE-DROPPING METHOD

Song [2004] reported that for a fixed computational effort, $\hat{p}_{RP}$ will be more efficient than $\hat{p}_{GD}$. The simulation results presented in Table IV essentially confirm these conclusions; namely that even using a small pool of gene-drop simulations to generate pseudo-replicate genome-scans yields a p-value estimate that is much more accurate. They also suggest that relative efficiency of the $\hat{p}_{RP}$ estimator increases dramatically for smaller p-values. For instance, in samples with 128 families and 20 initial replicates, relative efficiency is fairly modest (RE = 11.68) for a lod score threshold of 2.0 (average $\hat{p}_{RP}$ = .45366), becomes somewhat more pronounced (RE = 93.88) for a threshold of 3.0 (average $\hat{p}_{RP}$ = .06223) and increases sharply (RE = 653.1) for a threshold of 4.0 (average $\hat{p}_{RP}$ = .00667). Interestingly, our results also suggest that for a sample containing families of similar size, only a few seed replicates (e.g. 20) are required when the number of families is relatively large (e.g. 128) while for smaller sample sizes, it appears desirable to have more seed replicates.

In practice, it can be helpful to bear in mind that these results are an upper bound—if fixed computational effort is implicit to the comparison, actual relative efficiency of p_{GD} and p_{RP} will depend on the underlying data set. This relationship can best be understood in terms of the computational efficiency parameter

$$R_{REP} = (\#\text{pseudo-replicates calculated}/\text{time period to add 1 gene -dropping replicate})$$

which measures the number of pseudo-replicates that can be calculated in the time required to generate and analyze one gene-dropping replicate. In our experience, this parameter can range from 10 to 100,000 or more. For data sets where R_{REP} is high (e.g. Abecasis, 2004b with R_{REP} = 40,821), the relative efficiency of p_{GD} and p_{RP} should be on the order of that presented in Table IV, since accumulation of even a few extra gene-dropping replicates will require almost as much computational effort as accumulating several hundred thousand pseudo-replicates. For data sets with more modest values of R_{REP}, relative efficiency of the two estimators (assuming fixed computational effort) will fall in more moderate ranges.

CONVERGENCE OF VARIANCE ESTIMATES IN GENOME SCAN DATA

Although $\hat{W}_{fc}$ has the advantage of conservatively estimating family weights and variance for $\hat{p}_{RP}$, the tradeoff is that both $\hat{W}_{fc}$ and STD_{RP} converge somewhat slowly and large numbers of pseudo-replicate samples are required to remove the upward bias in $\hat{W}_{fc}$ and therefore STD_{RP}. Figure 2 illustrates the convergence behavior of

TABLE IV. Relative efficiency of p_{RP} and p_{GD} for family samples with 16–128 families, 20 or 50 replicates in each replicate pool

		20 initial replicates		50 initial replicates	
16	2.0	.65774	15.05	.65799	21.74
	2.5	.22973	29.48	.23108	44.86
	3.0	.04118	127.74	.04171	155.63
	3.5	.00292	912.88	.00299	1074.14
	4.0	.00008	N/A	.00008	N/A
32	2.0	.53024	22.48	.52550	31.56
	2.5	.24272	47.13	.23911	42.18
	3.0	.09898	95.43	.09724	100.82
	3.5	.03741	152.19	.03668	268.18
	4.0	.01292	403.82	.01260	666.64
64	2.0	.46819	30.49	.46054	14.28
	2.5	.18948	41.23	.18350	23.89
	3.0	.06728	96.42	.06414	44.85
	3.5	.02284	312.16	.02133	117.47
	4.0	.00771	465.60	.00697	401.69
128	2.0	.45366	11.68	.45482	19.79
	2.5	.17926	37.99	.17951	31.70
	3.0	.06223	93.88	.06221	87.20
	3.5	.02053	180.71	.02042	294.34
	4.0	.00667	653.10	.00665	350.50

STD_{RP} with increasing number of pseudo-replicates for the values of $\hat{p}_{RP}$ reported in Table I corresponding to family configuration 1 (2 children + parental genotypes) and a 10 cM microsatellite map. For each graph, the line in red marks the sample standard deviation and blue marks the estimated standard deviation. For these particular p-values ($p = .04461$, .01468, and .00472), convergence required several hundred thousand replicates. Furthermore, for all settings considered, the asymptotic limit of STD_{RP} conservatively estimated the sample standard deviation. In a practical sense, we recommend performing initial screens of all hypotheses of interest using a moderate number of pseudo-replicates (10,000–50,000) followed by additional simulations for promising hypotheses to refine variance estimates until confidence bounds are narrowed sufficiently for purposes of inference.

DISCUSSION

When compared to the conventional gene-dropping method for evaluation of genome-wide linkage peak p-values, the replicate pool method has several distinct advantages. First and foremost is computational efficiency. As we have shown, this method can be tremendously more efficient than gene-dropping for data sets that include one or more large families. When data include families with 20 or more individuals, the replicate pool method may require only a few hours to estimate a p-value that may require weeks or years to estimate using the traditional gene-dropping method. In the past, a typical solution to this problem has been to report an empirical p-value based on a few dozen gene-dropping replicates. While this is a reasonable approach given the lack of alternatives, it leaves the investigator in the ironic situation of being unable to assess significance in exactly the most interesting data sets. The replicate pool method represents a natural solution to this problem; many linkage packages calculate and report family-specific z-scores and for relatively small computational effort, a p-value that more accurately reflects the structure of the data set can be reported. We recommend that investigators generate a modest number of seed replicates for each of the families in their genome scan (say 50–100) and then use these to construct a very large number of pseudo-simulated genome scans (say 50,000–1,000,000) which can be analyzed in a computationally inexpensive manner.

Up to this point, the replicate pool has carried a potential caveat—as we have mentioned, the distribution of $\hat{p}_{RP}$ is difficult to characterize for any specific example. The simulation work we have done here suggests that the distribution of $\hat{p}_{RP}$ is an extremely good approximation for $\hat{p}_{GD}$. That said, one of the difficulties of this experiment was the sheer scale of computation involved—for instance, the results presented in Table I represent roughly 300 days of computer time. This made it challenging to obtain even relatively modest samples from each distribution. Further simulation work might be needed to determine if sampling conditions exist for which the replicate pool method does not perform as well as we have seen.

We have also demonstrated that for most data sets, the replicate pool p-value estimator is more efficient than the corresponding gene-dropping estimator when a fixed computational effort is assumed. For data sets where the replicate pool is most computationally efficient, we have shown that $\hat{p}_{RP}$ can be $>1{,}000$-fold more efficient than the traditional estimator for extremely significant results ($p < .002$) and 100–500 times more efficient for moderately significant results ($.01 < p < .05$). Although an analytical expression for calculating the variance of $\hat{p}_{RP}$ for an arbitrary linkage statistic is not available, we have presented an estimator that can be used to place a conservative bound on this value. Because

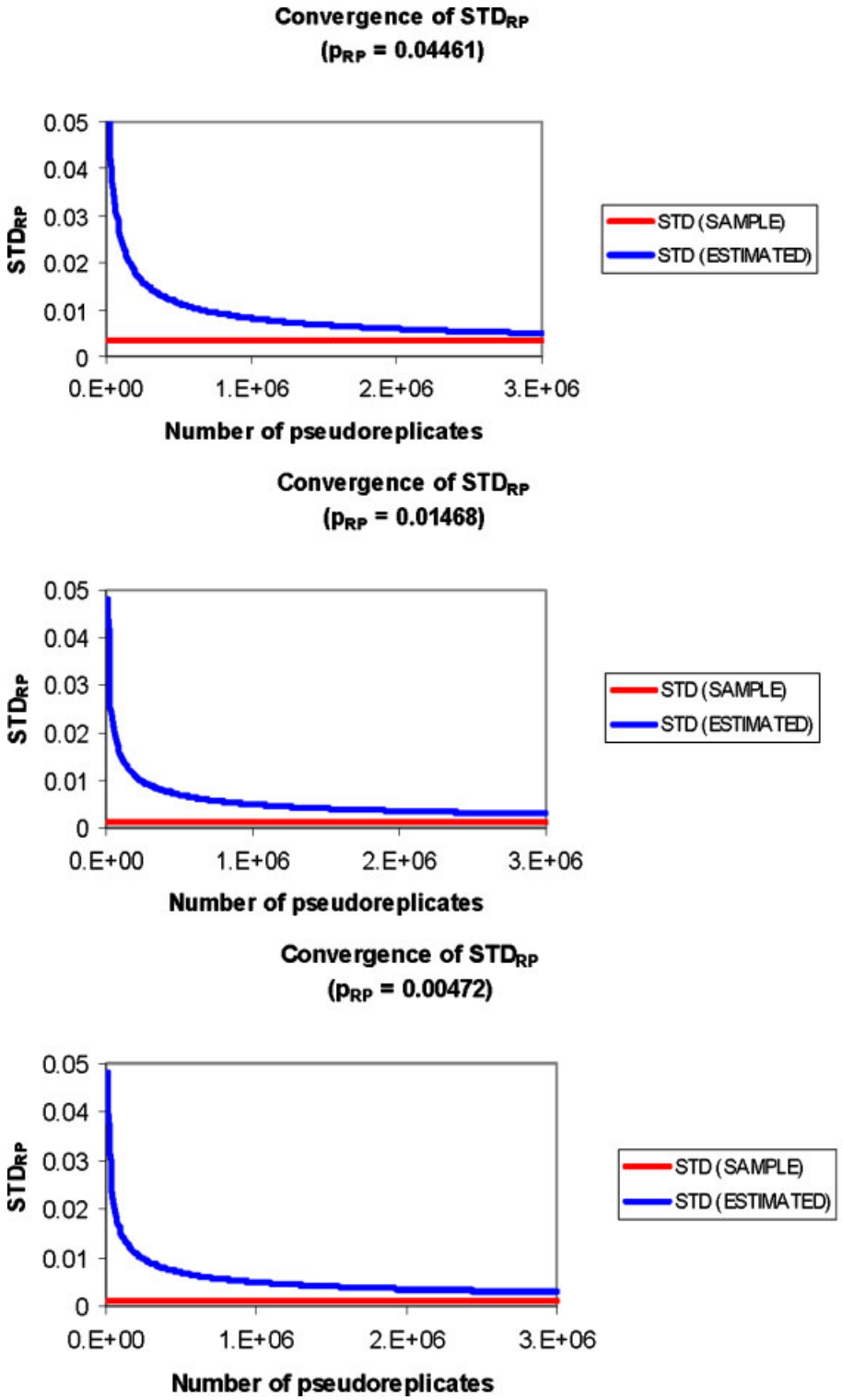

Fig. 2. Convergence of STD_{RP} for standard deviations in Table I, 10 cM microsatellite map, 2 children+parents genotyped.

this estimator tends to converge somewhat slowly, we recommend that promising hypotheses first be identified by a set of initial tests using a moderate number of pseudo-replicates (~50,000) and followed up with a second set of simulations if narrower confidence intervals for $\hat{p}_{RP}$ are desired.

The efficiency of the replicate pool method also makes it a good solution to the problem of properly correcting for multiple comparisons when linkage analyses are done on multiple correlated or nested outcomes. A typical example of this situation would be data presented in our applied example [Abecasis et al., 2004a]. Here, the phenotype of interest is notoriously difficult to define and the investigators have (quite reasonably) chosen to analyze several nested definitions of schizophrenia in an effort to identify genetic components of the syndrome. Once a set of interesting linkage peaks has been identified, the problem then is to correct for multiple comparisons while still accounting for correlation among outcome variables. Traditional gene-dropping would be a logical approach to solving this problem, but can very quickly become computationally prohibitive. Because the replicate pool method is so much more computationally efficient than gene-dropping, it can be used to test any number of complex hypotheses in a reasonable time frame and should be a useful tool for approaching this problem.

Yet another consideration for the replicate pool method that deserves further evaluation is the performance of $\hat{p}_{RP}$ in heterogeneous family samples. Although $\hat{p}_{RP}$ behaved well in both the simulated heterogeneous family sample and the schizophrenia data set we considered, it is possible that sampling scenarios may exist where the performance of $\hat{p}_{RP}$ might degrade because a small subsample of families is disproportionately influential and not well represented in the initial replicate pool. When this is anticipated, we recommend that different numbers of seed replicates be used for each family. For example, 20–50 replicates could be used for the smaller less informative pedigrees, and additional replicates could be generated for larger pedigrees that more greatly influence the final linkage signal. Our software implementation allows for different numbers of seed replicates to be provided as input for each family.

We have implemented the replicate pool method along with our variance estimator in the Pseudo package. This implementation is able to consider one or more phenotypes simultaneously and evaluate significance for a single peak or for a group of linkage peaks. The package and C++ source code are freely available from our website at http://www.sph.umich.edu/csg/abecasis/pseudo. Pseudo is designed to work with the linkage package Merlin, which can be used to generate multiple null replicates of a data set and save the associated family-specific z-scores for further sampling by Pseudo.

ACKNOWLEDGMENTS

The authors thank Andrew Skol for many helpful comments and suggestions.

ELECTRONIC INFORMATION

The replicate pool method for p-value estimation is implemented in Pseudo, a freely available C++ program which works with the linkage package Merlin. Source code, executables and a brief tutorial are available on our web site at http://www.sph.umich.edu/csg/abecasis/pseudo/. Supplemental information describing development of our estimator for family variance weights is available at http://www.sph.umich.edu/csg/abecasis/pseudo/supplement/

REFERENCES

Abecasis GR, Cherny SS, Cookson WO, Cardon LR. 2002. Merlin-rapid analysis of dense genetic maps using sparse gene flow trees. Nat Genet 30:97–101.

Abecasis GR, Burt RA, Hall D, Bochum S, Doheny KF, Lundy SL, Torrington M, Roos JL, Gogos JA, Karayiorgou M. 2004a. Genomewide scan in families with schizophrenia from the founder population of Afrikaners reveals evidence for linkage and uniparental disomy on chromosome 1. Am J Hum Genet 74:403–417.

Abecasis GR, Yashar BM, Zhao Y, Ghiasvand NM, Zareparsi S, Branham KEH, Reddick AC, Trager EH, Yoshida S, Bahling J, Filipova E, Elner S, Johnson MW, Vine AK, Sieving PA, Jacobson SG, Richards JE, Swaroop A. 2004b. Age-related macular degeneration: a high-resolution genome scan for susceptibility loci in a population enriched for late-stage disease. Am J Hum Genet 74:482–494.

Abecasis GR, Wigginton JE. 2005. Handling marker-marker linkage disequilibrium: pedigree analysis with clustered markers. Am J Hum Genet 77:754–767.

Bacanu S. 2005. Robust estimation of critical values for genome scans to detect linkage. Genet Epidemiol 28:24–32.

Besag J. 1991. Sequential Monte Carlo p-values. Biometrika 78: 301–304.

Elston RC. 1998. Methods of linkage analysis—and the assumptions underlying them. Am J Hum Genet 63:931–934.

Faraone SV, Skol AD, Tsuang DW, Young KA, Haverstock SL, Prabhudesai S, Mena F, Menon AS, Leong L, Sautter F, Baldwin C, Bingham S, Weiss D, Collins J, Keith T, VandenEng JL, Boehnke M, Tsuang MT, Schellenberg GD. 2005. Genome scan of schizophrenia families in a large veterans affairs cooperative study sample: evidence for linkage to 18p11.32 and for racial heterogeneity on chromosomes 6 and 14. Am J Med Genet B Neuropsychiatr Genet 139:91–100.

Kong A, Cox NJ. 1997. Allele-sharing models: LOD scores and accurate linkage tests. Am J Hum Genet 61:1179–1188.

Kruglyak L, Daly M, Reeve-Daly MP, Lander ES. 1996. Parametric and nonparametric linkage analysis: a unified multipoint approach. Am J Hum Genet 58:1347–1363.

Kruglyak L, Daly MJ. 1998. Linkage thresholds for two-stage genome scans. Am J Hum Genet 62:994–996.

Lander E, Kruglyak L. 1995. Genetic dissection of complex traits: guidelines for interpreting and reporting linkage results. Nat Genet 11:241–247.

McPeek MS. 1999. Optimal allele sharing statistics for genetic mapping using affected relatives. Genet Epidemiol 16:225–249.

Morton NE. 1955. Sequential tests for the detection of linkage. Am J Hum Genet 7:277–318.

Morton NE. 1998. Significance levels in complex inheritance. Am J Hum Genet 62:690–697.

Ott J. 1989. Computer-simulation methods in human linkage analysis. Proc Natl Acad Sci USA 86:4175–4178.

Sawcer S, Jones HB, Judge D, Visser F, Compston A, GoodFellow PN, Clayton D. 1997. Empirical genomewide significance levels established by whole genome simulations. Genet Epidemiol 14:223–229.

Song KK, Weeks DE, Sobel E, Feingold E. 2004. Efficient simulation of p values for linkage analysis. Genet Epidemiol 24:1–9.

Teng J, Siegmund D. 1997. Combining information within and between pedigrees for mapping complex traits. Am J Hum Genet 60:979–992.

Terwilliger JD, Ott J. 1992. A multisample bootstrap approach to the estimation of maximized-over-models lod score distribution. Cytogenet Cell Genet 59:42–144.

Terwilliger JD, Ott J. 1994. Handbook of Human Genetic Linkage. Baltimore: Johns Hopkins University Press.

Whittemore AS, Halpern J. 1994. A class of linkage tests using affected pedigree members. Biometrics 50:118–127.

Wiltshire S, Cardon LR, McCarthy MI. 2002. Evaluating the results of genomewide linkage scans of complex traits by locus counting. Am J Hum Genet 71:1175–1182.