JOURNAL OF COMPUTATIONAL BIOLOGY
Volume 16, Number 5, 2009

Pp. 659–676
DOI: 10.1089/cmb.2008.0175

Fast and Accurate Calculation of a Computationally Intensive Statistic for Mapping Disease Genes

SANG-CHEOL SEOK,[1] MICHAEL EVANS,[3] and VERONICA J. VIELAND[1,2]

ABSTRACT

Many statistical methods in biology utilize numerical integration in order to deal with moderately high-dimensional parameter spaces without closed form integrals. One such method is the PPL, a class of models for mapping and modeling genes for complex human disorders. While the most common approach to numerical integration in statistics is MCMC, this is not a good option for the PPL for a variety of reasons, leading us to develop an alternative integration method for this application. We utilize an established sub-region adaptive integration method, but adapt it to specific features of our application. These include division of the multi-dimensional integrals into three separate layers, implementing internal constraints on the parameter space, and calibrating the approximation to ensure adequate precision of results for our application. The proposed approach is compared with an empirically driven fixed grid scheme as well as other numerical integration methods. The new method is shown to require far fewer function evaluations compared to the alternatives while matching or exceeding the best of them in terms of accuracy. The savings in evaluations is sufficiently large that previously intractable problems are now feasible in real time.

Key words: complex traits, linkage and LD analyses, posterior probability of linkage, numerical integration, sub-region adaptive method.

1. INTRODUCTION

THE PPL STATISTIC has been developed as a method for the rigorous accumulation of evidence for or against linkage and/or linkage disequilibrium (LD) in human pedigrees (Vieland, 1998), for use in localizing disease genes on a map of the human genome and understanding their underlying relationships. The method is quite general in terms of pedigree structures (case-control data, affected sib pairs, general pedigrees); marker data types (microsatellites, SNPs, two-point and multi-point calculations); genomic features (e.g., sex-specific recombination); and in particular, in its handling of a very general class of ''trait models,'' or models relating observed phenotypes to underlying unobserved genotypes. These trait models currently handle dichotomous traits, quantitative traits, and both types of traits together in the same pedigree

[1]Battelle Center for Mathematical Medicine, The Research Institute at Nationwide Children's Hospital, Columbus, Ohio.
[2]Departments of Pediatrics and Statistics, The Ohio State University, Columbus, Ohio.
[3]Department of Statistics, University of Toronto, Toronto, Ontario, Canada.

using a quantitative trait threshold model; current implementations also handle two-locus interaction models. For an overview and relevant literature, see Vieland (2006).

At the heart of all forms of the PPL is the calculation of multi-dimensional integrals, over two classes of parameters. First, the PPL is computed from an underlying likelihood, which is parameterized in terms of a vector of unknown parameters of the trait model (including such things as gene frequencies), which are integrated. The result of this integration is called the Bayes ratio (BR), and it is a function of the remaining localization parameters: in the case of linkage mapping, the recombination fraction, θ, which measures distance along each chromosome in recombination units or centiMorgans; and in the case of LD mapping, the standardized LD parameter D', which measures distance in the far smaller LD units. The second level of integration is then carried out across θ and/or D'. One reason to separate the integration into these two levels is that we need to retain the BRs obtained from a given data set for use in Bayesian sequential updating over subsequent data sets (Huang and Vieland, 2001; Vieland et al., 2001), which requires storing the output as a function of θ and/or D' for future use, necessitating integration over trait parameters in a separate step. A second reason to conduct the integration in stages is that the priors are handled differently for the two classes of parameters, with uniform priors used in calculation of the BR itself, but priors involving some discontinuities used for localization parameters (Logue and Vieland, 2005a).

Genetic likelihoods are generally computed algorithmically, for instance, using the Elston-Stewart (Elston and Stewart, 1971) algorithm to traverse the pedigree. We have re-implemented the Elston-Steward algorithm in a form that uses the algorithm to generate and store the likelihood as a complex polynomial in the parameters of the model (Wang et al., 2007a, b). The degree of the resulting polynomial depends on the number of generations in the pedigree, the number of individuals in the family, and other features of the data, and it can be quite high especially in larger pedigrees. Moreover, the PPL itself is actually computed from a ratio of polynomials. Thus, analytic integration is not available, necessitating numerical approaches. On the other hand, storing the likelihood in polynomial form allows us to efficiently evaluate it across large numbers of parameter values as is needed for numerical integration (NI).

Thus, we have both a need for NI and a software platform in which to efficiently implement it, but this leaves open the question of which if any existing NI methods will be optimal for our particular problem, where what is optimal involves a balance between speed and accuracy. In this article, we present an approach to NI that is based on a sub-region adaptive method, called DCUHRE (Berntsen et al., 1991). DCUHRE allows us to deal effectively with the different complexities of parametric sub-regions and to guarantee the exactness of the approximation up to some level. We adapt DCUHRE by separating the integration into layers, implementing internal constraints as dictated by the genetic models, and calibrating for the desired degree of accuracy. We also evaluate the speed of the resulting method and show that it makes even large-scale problems computationally feasible.

2. METHODS

In this section, we first introduce genetic linkage analysis and present the underlying mathematics of the PPL. This provides the context for our application of NI. We then survey some alternative approaches to NI for multi-dimensional problems as they would apply to calculation of the PPL. Finally, we describe the simulation procedures used below in comparative evaluation of alternative NI methods for computing the PPL, as well as an application to a real data set.

2.1. Overview of linkage analysis using the PPL

Linkage analysis is a technique for localizing a putative disease-predisposing gene on a map of the human genome. The technique is based on the biological phenomenon of recombination, whereby homologous chromosomes exchange genetic material during meiotic cell division. The probability of this exchange between any two chromosomal locations is proportional to the distance between them, and this distance is measured by the recombination fraction θ, defined as the probability of a recombination event (observable exchange of material, or ''crossover'') between the locations. Linkage analysis works by beginning with a set of families (parents, children, and possibly grandparents, cousins, etc.), a marker map (set of polymorphisms, or sites exhibiting genetic variation within the population, distributed across the genome and amenable to molecular assay), and a set of phenotypes (e.g., ''affected'' or ''unaffected'') for the family members. Statistical methods are then used to systematically model the relationship between segregation of

each genomic marker or location in turn and segregation of the disease within the families. When the disease gene is far from the marker location, $\theta = 1/2$; but if the disease gene is nearby on the same chromosome, $\theta < 1/2$, with very small values of θ indicating very short genetic distances between trait and marker. This enables us to "map" the disease gene, without knowing anything about its structure or function ahead of time, based solely on positional information.

A common statistical technique in linkage analysis is to work with the likelihood for the pedigree. This is, for instance, the basis of the familiar LOD score, which is the logarithm of the ratio of the likelihood in θ divided by the likelihood with θ fixed at 1/2 (which represents "no linkage"). The likelihood in its general form involves the location parameter θ, as well as a set of parameters governing the relationship between the observed phenotypes and the unobserved underlying genotypes at the disease locus. For all but the simple Mendelian disorders, these parameters are unknown, and indeed the exact structure of the likelihood (including the number of trait parameters and their interrelationships) is unknown. However, a large body of literature in statistical genetics supports the use of robust approximating likelihoods for purposes of linkage analysis, which can be expressed as a function of a relatively small number of parameters. One such standard parameterization includes a disease gene frequency and a set of three penetrances (one for each possible disease genotype, where the penetrance is defined as the probability of being affected given the genotype). A frequently used extension of this likelihood includes an admixture parameter α, which is defined at the level of the whole pedigree and represents the probability that the pedigree is of the "linked" versus "unlinked" type (Smith, 1959). Inclusion of α in the likelihood allows for the common phenomenon of locus heterogeneity, whereby a single clinical disease can be caused by different underlying genes in different families. For additional details, see Ott (1999). Written in terms of the LOD as defined above, inclusion of α leads to Heterogeneity LOD (HLOD), defined as

$$HLOD = \sum_{pedigrees} \log_{10} [\alpha 10^{LOD} + (1 - \alpha)]. \tag{1}$$

For computational reasons, we use the HLOD as just defined as the basis for computation of the PPL itself (Vieland, 1998).

The formula for the basic form of the PPL is easily expressed in terms of an integrated likelihood ratio, BR, as a function of θ,

$$BR(\theta) = \int_{\alpha} \int_{g} 10^{HLOD(\theta, \alpha, g)} \pi(\,\cdot\,) dg d\alpha \tag{2}$$

or with polynomial expressions

$$\int_{\alpha} \int_{g} \prod_{i} (\alpha \frac{P_i(g, \theta)}{P_i(g, 0.5)} + 1 - \alpha) \pi(\,\cdot\,) dg d\alpha, \tag{3}$$

where $\pi(\cdot)$ is a probability distribution on α and g, and P_i is the polynomial for the $pedigree_i$. In order to give some idea of the complexity of these polynomials, Figure 1 shows the resulting equation for one of the simplest possible pedigree configurations, a nuclear family with two children; polynomial representations of the likelihood for more complicated pedigrees can be far more complex than this. While a uniform distribution is typically assumed on α and g, a skewed distribution is applied for θ and D'. The trait parameter vector g includes the gene frequency gf and penetrances (f_{DD}, f_{Dd}, f_{dd}) as described above. That is, the BR is the integral of the anti-log of the HLOD over all parameters except θ. For the basic dichotomous trait model discussed here, all five trait parameters are probabilities and therefore have ranges between 0 and 1.

Integrating θ itself out of the BR, we obtain

$$I_{LE} = \int_{\theta \in [0, 0.5)} BR(\theta) \Pr(\theta | L) d\theta, \tag{4}$$

where $\Pr(\theta|L)$ represents the conditional distribution of θ given linkage. This distribution is skewed toward small values (e.g., we overweight $0 \leq \theta < 0.05$). Finally, the PPL, or posterior probability of linkage, can be computed via Bayes' theorem as

$$
\begin{aligned}
&p(f,x,y,z,\theta)\\
&= 0.08f^2x(0.005f^2x^2(1-x)\\
&\quad +0.16(1-f)f(1-y)x(\frac{y(1-\theta)+x\theta}{4})(\frac{y\theta+x(1-\theta)}{4})\\
&\quad +0.005(1-f)^2y^2(1-z))\\
&+ 0.08f(1-f)y(0.08f^2(1-x)(\frac{y(1-\theta)+x\theta}{4})^2\\
&\quad +0.16f(1-f)(1-y)(\frac{z(1-\theta)^2+2y\theta(1-\theta)+x\theta^2}{4})(\frac{z\theta(1-\theta)+y(1-\theta)^2+y\theta^2+x\theta(1-\theta)}{4})\\
&\quad +0.08(1-f)^2(1-z)(\frac{z(1-\theta)+y\theta}{4})^2)\\
&+ 0.08f(1-f)y(0.08f^2(1-x)(\frac{y\theta+x(1-\theta)}{4})^2\\
&\quad +0.16f(1-f)(1-y)(\frac{y(1-\theta)^2+y\theta^2+z\theta(1-\theta)+x\theta(1-\theta)}{4})(\frac{2y\theta(1-\theta)+z\theta^2+x(1-\theta)^2}{4})\\
&\quad +0.08(1-f)^2(1-z)(\frac{y(1-\theta)+z\theta}{4})^2)\\
&+ 0.08(1-f)^2z(0.005f^2y^2(1-x)\\
&\quad +0.16f(1-f)(1-y)(\frac{y\theta+z(1-\theta)}{4})(\frac{y(1-\theta)+z\theta}{4})\\
&\quad +0.005(1-f)^2z^2(1-z)))
\end{aligned}
$$

FIG. 1. Example of a very simple polynomial for a family of four, including parents and two affected children, with gene frequency, f, penetrance vectors, (x,y,z), and the recombination fraction θ.

$$PPL = \frac{P(L)I_{LE}}{P(L)I_{LE} + (1 - P(L))}, \tag{5}$$

where P(L) is the prior probability of linkage. P(L) is usually set to 2% in accordance with Elston and Lange (1975). The PPL ranges from 0 to 1, with values of $>2\%$ representing evidence in favor of linkage and values of $<2\%$ representing evidence against linkage.

The PPL has already been applied to a number of real human genetic data sets (Logue et al., 2003, 2006; Bartlett et al., 2004, 2005; Bartlett and Vieland, 2005, Wang et al., 2001) with striking results, although computational issues such as those addressed in this paper still impede widespread application.

One important extension of this basic form of the PPL will also be considered here. A second genomic feature that can be used for gene mapping is LD. LD occurs when two loci are so close to one another that recombination between them is for all practical purposes negligible, leading to an association at the population level between variants at the loci. (Whereas linkage refers to co-segregation of variants within families, LD refers to associations of variants in the general population.) Again, see Ott (1999) for details. In the presence of LD, evidence for linkage can be enhanced by explicitly allowing for LD in the linkage likelihood; additionally, LD is of interest in its own right. We have therefore generalized the PPL to allow for LD by including the standardized LD parameter D' in the likelihood, leading to a PPL variant we call the LD_PPL (Yang et al., 2005). This adds an additional localization parameter to the likelihood, but does not affect the parameterization of the trait model g. Thus, we have the LD integral

$$I_{LD} = \int_{D' \in [-1,1]} \int_{\theta \in [0, 0.5)} BR(D', \theta) \Pr(\theta, D'|L) d\theta dD'. \tag{6}$$

And the LD_PPL is defined as

$$LD\_PPL = \frac{P(L)I_{LD}}{P(L)I_{LD} + (1 - P(L))}. \tag{7}$$

Thus, the calculation of the PPL can be broken down in terms of either the 1-dimensional integration over θ or the 2-dimensional integration over θ and D', each of which in turn entails the calculation of the multi-dimensional integrals, or Bayes ratios, over the trait parameters. While extensions of the PPL and LD_PPL exist that involve somewhat different trait parameter spaces (e.g., to handle quantitative traits [Bartlett, 2007]),

in this article we focus on accurate numerical integration as needed for this basic form of the PPL and LD_PPL, as a basis for handling all forms.

The behavior of the integrals in θ and D' is smooth, and as just described, this step is handled separately from the integration involved in calculation of the BR, where most of the NI "action" takes place. In selecting an NI method, therefore, we focus primarily on accuracy and efficiency in calculating the BR; reserving calculation of the PPL for separate discussion.

2.2. *Multi-dimensional numerical integration*

The calculation of multi-dimensional integrals is generally a very difficult problem. Typically, the difficulty rises with dimension but even in relatively low dimensions the problem can be difficult when we have very little information about the integrand of interest. While there are a variety of general methods available for approximating such integrals, many problems require the design of an algorithm that is tailored to the specific integrand. In our application, each polynomial is not really of very high dimension but things are complicated by the fact that we are multiplying together so many functions to get the integrands. Here, we survey numerical integration (NI) techniques that could be used for calculation of the PPL. For additional NI methods, see Evans and Swartz (2000).

2.2.1. Fixed grid schemes. Fixed grid schemes have the advantage of simplicity. In the calculation of the PPL, this approach simply evaluates the BR at the pre-selected parameter values and averages them to estimate integrals. In earlier work, we had developed a standard fixed grid, which appeared to have good empirical properties based on extensive simulations. The simplicity of this fixed grid scheme made it easy for us to develop a visualization tool to explore the likelihood space as well (Park et al., 2006). This grid included six values for the gene frequency (*gf*) (0.001, 0.01, 0.1, 0.3, 0.5, and 0.8); 275 (f_{DD}, f_{Dd}, f_{dd}) combinations of penetrances, satisfying certain ordering constraints, with 0.1 step-size ranging from 0 to 0.9, and 0.999; 51 θ values on [0, 0.5]; and 20 α values from 0.05 to 1.0. All variables except *gf* are evenly spaced. This grid system requires $6*275*20 = 33{,}000$ function evaluations to calculate the $BR(\theta)$ for a given θ. Thus, the calculation of the PPL requires about 1.7 million function evaluations for a single marker or genetic position.

Based on extensive simulations, we ascertained that there are no irregular or sharp changes in the integrands between adjacent values of *gf*, α and θ, but that the same is not always true for the penetrance variables (f_{DD}, f_{Dd}, and f_{dd}). We therefore introduced an additional "fine grid" scheme, which uses a 0.01 step-size for each of the three penetrance variables, producing approximately a quarter million combinations of penetrance variables. In order to decrease the computational time, θ is coarsened to a step-size of 0.05 (we confirmed empirically that this reduction in step-size did not affect the results). The total number of function evaluations for this fine grid scheme is therefore about 330 million points. While the computational results from this fine grid scheme are used as the basic "gold standard" for assessing the efficacy of other approaches, finer grids than this one are used as needed in what follows. Note that in the standard grid *gf* actually overweights a small interval close to 0 by sampling more values around 0. This overweighting is avoided and replaced with evenly spaced values in the finest grid.

2.2.2. Markov Chain Monte Carlo (MCMC). While not developed as an NI technique per se, in statistical genetics MCMC is the only widely used approach when NI is required. Thus it might appear to be the obvious choice in our application as well. However, the development of a suitable MCMC algorithm depends on characteristics of the integrand, and given the particular complications of our integrand, it may not be possible to find a useful sampler. For example, there does not appear to be an easy way to implement Gibbs sampling for our problem. If we had no alternatives, we would perhaps have no choice but to invest time and effort in an attempt to develop appropriate MCMC samplers. However, there are alternatives, which we consider in detail below.

2.2.3. Monte Carlo algorithms. For multi-dimensional integrals, Monte Carlo (MC) algorithms have been very popular. There are a wide variety of MC algorithms available, involving uniform sampling or importance sampling, as well as many variance reduction techniques. MC algorithms are based on using pseudorandom numbers to approximate the integrals. We have considered uniform sampling and two MC algorithms with importance sampling. The first importance sampling algorithm we considered is MC VEGAS (Lepage, 1978), which is a product importance sampler that attempts to find the optimal such sampler by

adapting to projections of the integrands along the individual axes. Of course, if the integrand cannot be well approximated by such a product then we can expect the approximation of the integral to be poor. The second is MC MISER (Press and Farrar, 1990), which is a sub-region adaptive algorithm and uses uniform sampling within sub-regions.

MC VEGAS goes through five iterations for importance sampling, where each iteration uses 2000 function calls. In what follows, we set MC MISER to start with 8,000 initial calls, but in practice it actually used around 4,500 function evaluations for each *BR(θ)* because of the parameter constraints.

2.2.4. Sub-region adaptive algorithm: DCUHRE. DCUHRE is an adaptive multi-dimensional integration method, which uses fully symmetric multiple quadrature rules (Berntsen et al., 1991). Each rule consists of points and weights. These are used to approximate integrals over an *n*-dimensional hypercube via a weighted sum of function evaluations at the points. The fully symmetric rules used in DCUHRE are an imbedded family of $2m+1$ degree rules generated by $m+1$ generators. (That is, all points in $2m-1$ degree rule are included in $2m+1$ degree rule.) The points and weights are determined by m and n. For example, when $m=4$ and $n=4$ (that is, a nine-degree rule on a four-dimensional hypercube), 153 points are used to approximate an integral over a hypercube. Each $2m+1$ degree rule of DCUHRE exactly integrates multivariable polynomials of degree up to $2m+1$. Like quadrature rules for one-dimensional space, the rule points and weights are generated by solving a very complicated nonlinear system of equations. These points and weights have been tabulated for many choices of m and n. The rules are most economical and useful for $3 \leq m \leq 6$ and $3 \leq n \leq 3m$ (Genz and Malik, 1983). DCUHRE currently uses 7-, 9-, 11-, and 13-degree rules.

For complex problems, instead of creating higher degree rules, DCUHRE uses a sub-region adaptive scheme as follows. Given a desired relative error of $\varepsilon > 0$ for the approximation, the algorithm starts with one hypercube region $R_{11} = [0,1]^n$. At the k-th step $[0,1]^n$ has been partitioned into k sub-regions. Rules are then applied in each sub-region to get estimates $I_{k1}, \ldots, I_{kk}$ of the corresponding integrals and estimates $E_{k1}, \ldots, E_{kk}$ of the absolute errors in each sub-region. The error estimate E_{ki} is obtained by computing I_{ki} using rules of different orders and comparing the results. As such using an imbedded family of rules is very convenient as this reduces the number of function evaluations. The algorithm stops if $E_{k1} + \ldots + E_{kk} < \varepsilon(I_{k1} + \ldots + I_{kk})$. Otherwise, the sub-region, R_{ki}, with the biggest absolute error estimate is bisected along a coordinate axes, so there are now $k+1$ sub-regions. DCUHRE selects the coordinate direction for the bisection, where the integrand is most variable. The estimates of the integrals and errors of the newly created sub-regions replace those of the bisected sub-region.

The advantage of the sub-region adaptive approach over the fixed grid approach is that it attempts to learn about the integrand iteratively, with the result that in many problems we can obtain accurate approximations with far fewer function evaluations. This is because the sub-region adaptive approach places points where significant contributions to the integral are made rather than distributing the points uniformly across the domain of integration. Additionally, little additional computation is required to obtain an error estimate as discussed above. As the dimension of the integrand goes up, DCUHRE should be used carefully. Tests for efficiency and reliability on various families of integrands show that DCUHRE works well for up to 15-dimensional integrals (Berntsen et al., 1988). For the calculation of the BR, which involves five-dimensional integration, we use the nine-rule as recommended.

In what follows, we set DCUHRE to execute a maximum of 10,000 function calls, and to terminate when the error estimate drops below 1% (see below for additional evaluation of these settings). When the integrand is almost flat, DCUHRE's error estimate is small, and the integration terminates after a small number of function calls.

One additional implementation detail is that, for genetic reasons, we generally impose a constraint on the penetrances:

$$f_{DD} \geq f_{Dd}, f_{Dd} \geq f_{dd}, \text{ and } f_{DD} \neq f_{Dd} \neq f_{dd}. \tag{8}$$

In the fixed-grid and MC approaches, we handle these constraints by checking all the points used for integration to see whether they satisfy the constraints, building combinations of the three penetrance variables which satisfy the constraints and evaluating function values for these combinations only. This is inefficient for the MC methods, because it wastes computation time checking the constraints (up to five out of six samplings will end up being discarded for MC algorithms), but it remains an accurate way to implement the constraints. However, when we consider quadrature rules, this approach destroys their exact

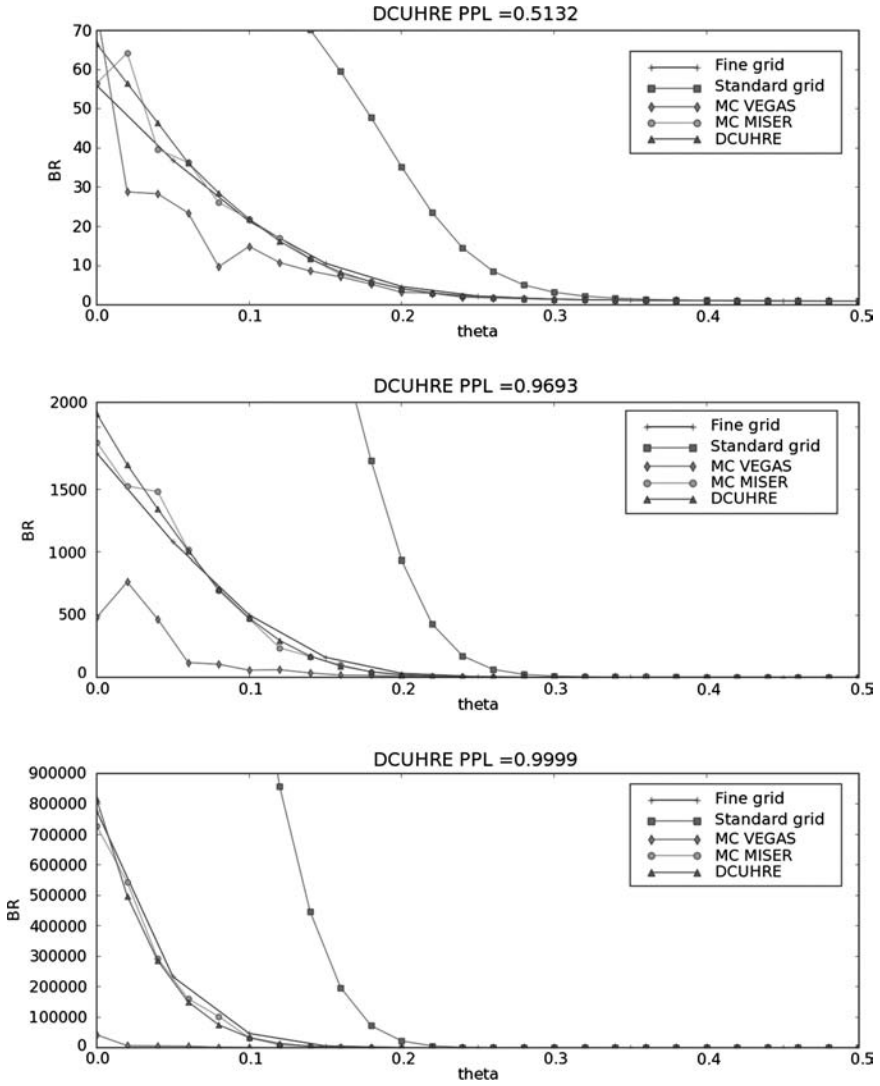

FIG. 2. BR's for three pedigree sets with medium, high, very high PPL's by five methods, fine grid in plus, standard grid in square, MC VEGAS in diamond, MC MISER in circle, and DCUHRE in triangle.

integration properties for polynomials. Therefore in implementing DCUHRE, we have handled the constraints by transforming the penetrance region into a hypercube. Using the variable transformation, $f_{DD} = u$, $f_{Dd} = uv$, $f_{dd} = uvw$, the integration is carried out on a hypercube domain $H = [0,1]^5$. Then Equation (2) for $BR(\theta)$ can be rewritten as

$$BR(\theta) = \int_H 10^{HLOD(\theta, \alpha, gf, u, uv, uvw)} \pi(\,\cdot\,) u^2 v dH, \tag{9}$$

where u^2v is the Jacobian of the transformation.

2.3. Simulated and real pedigree data for evaluating numerical integration methods

In order to compare alternative NI methods in this application, we needed data representing a wide range of BRs, to ensure that results apply across the spectrum of results we might obtain in real applications. The use of simulation for such purposes is routine in statistical genetics, and gives rise to data having the same characteristics as real data, that is, such that if the method can be applied to the simulated data it can be applied to real data without further modification. For this reason we relied on simulated data for primary

evaluation of the different NI approaches. We simulated 1200 replicates of data sets containing either 30 very small (four person), 30 medium-sized, or 30 very large (>4 generation) pedigrees; generated under conditions of locus heterogeneity, at both ''linked'' and ''unlinked'' genetic markers. The range of generating conditions was designed to provide a large range of BRs as well as to allow examination of the effects of pedigree structure. (Because different types of pedigree contain different amounts of information regarding certain underlying genetic parameters, it could be that NI approaches behave differently with different pedigree structures; however, this turned out not to be the case, and the issue of pedigree structure is not further discussed here.) In order to consider the LD_PPL, an additional 54 data sets were generated using similar initial conditions, but also introducing the LD parameter D' into the model.

In addition, we wanted to compare methods with an application to real data. For this purpose, we chose an autoimmune thyroid disease (AITD) data set, described in detail elsewhere (Tomer et al., 2003; Vieland et al., 2008). In brief, the data set comprises 102 pedigrees, ranging from four-person nuclear families to some large, three-generational pedigrees, each of which has two or more individuals with AITD. Marker data were available genome-wide (based on an approximate 10-cM spacing of microsatellite markers), and we computed three-point (multipoint) PPLs across the genome.

3. RESULTS

We divide the results into three subsections. First, we consider comparative accuracy for the standard grid, MC VEGAS, MC MISER, and DCUHRE methods in comparison with the fine grid, in application to calculation of the BR. As shown below, this leads us to select DCUHRE as the method of choice going forward. Second, we then present several refinements of the DCUHRE implementation that lead to greater efficiency without affecting its accuracy, including the use of error analysis to ensure sufficient accuracy in calculation of the PPL itself. Third, and finally, we consider some practical adaptations of DCUHRE and illustrate them with application to a real data set.

3.1. Comparative accuracy of NI methods in application to the BR

In order to illustrate the relationship between the BR and the PPL, we selected three data sets representative of ''medium,'' ''high,'' and ''very high'' PPLs respectively, as shown in Figure 2. Because by definition, the BR converges to 1 as θ moves to 0.5, and due to the usual prior on θ which ''up-weights'' BRs over small values, BRs near $\theta = 0$ are the most influential on the final PPL. Focusing particularly on small values of θ, there are three points to note regarding Figure 2. First, MC MISER and DCUHRE both appear to agree closely with the fine grid and with one another, while MC VEGAS deviates from the other two by returning smaller BRs, and the standard grid deviates by returning (considerably) higher BRs. We defer additional consideration of the standard grid for separate discussion, below.

Second, the two MC algorithms (MC VEGAS and MC MISER) show some inconsistency, presumably due to their reliance on random numbers. For instance, MC MISER has both concave up and concave down segments around $\theta = 0$ in the first two graphs, with a zig-zag pattern in the first graph. But when the BR maximizes at $\theta = 0$ (as is the case in all three graphs shown here), it is known that the BR itself uniformly decreases in concave down manner as θ increases.

Third, Figure 2 illustrates that ordinary data can give rise to an extremely large range of BRs. The form of the PPL, however, means that the required accuracy in different parts of the range will differ. For instance, when the BR is >1000 over small values of θ, the PPL is generally very close to 1; therefore achieving a high degree of accuracy above that range is less important than it is in the range $10.0 < \mathrm{BR} \leq 100$, where small differences in the BR can correspond to larger differences in the PPL.

With these points in mind, Figure 3 shows this same comparison across methods for the full set of 1200 simulated replicates, restricting attention to $\theta = 0$. As can be seen, it remains generally true that DCUHRE and MC MISER are consistently close to one another and to the fine grid, while MC VEGAS tends to differ, systematically returning smaller values of the BR across the full BR range.

This suggests that DCUHRE and MC MISER are more accurate than MC VEGAS, however, in order to confirm that it is not MC VEGAS which is returning the correct BR's we performed the following experiment. We divided the simulated data sets into four categories: Very Low ($\mathrm{BR} \leq 3$), Low ($3 < \mathrm{BR} \leq 10.0$), Medium ($10.0 < \mathrm{BR} \leq 100$), and High ($100 < \mathrm{BR} \leq 1000$). (Recall that accuracy is far less of an issues for

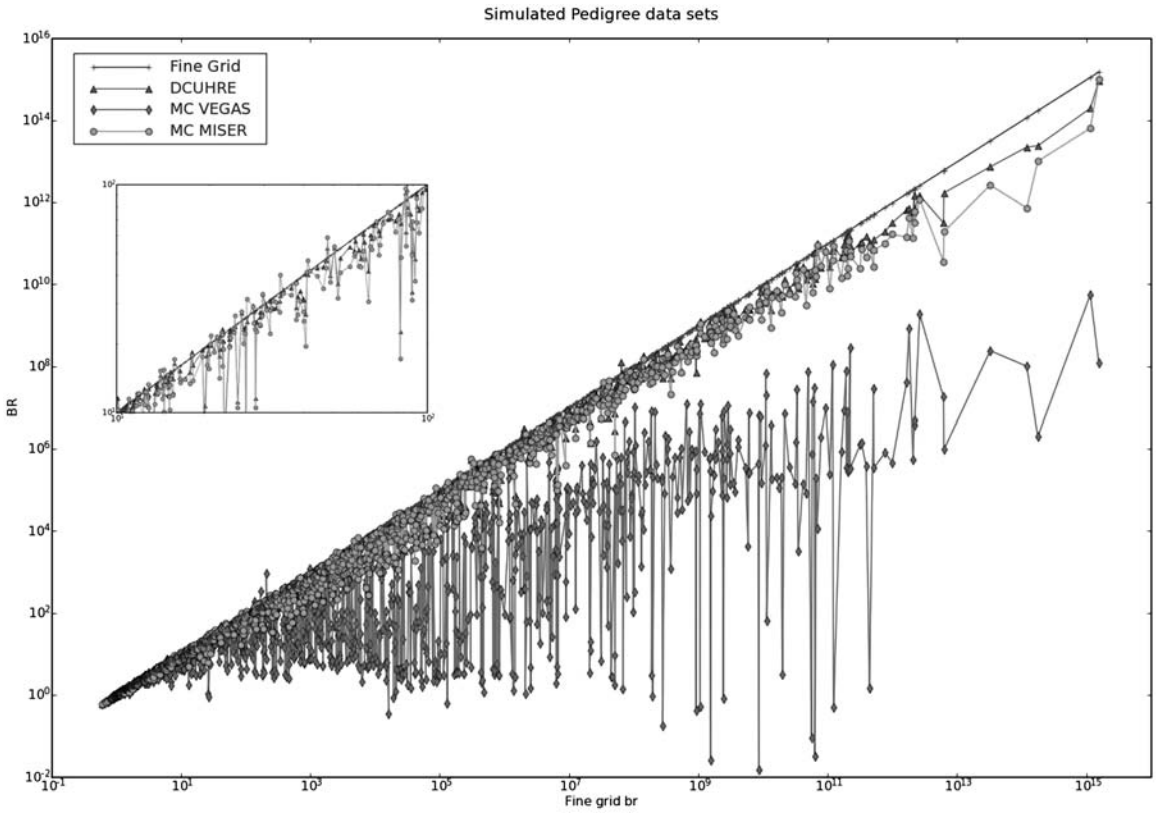

FIG. 3. Comparison of three different NI methods, DCUHRE, MC MISER, and MC VEGAS, with the fine grid on 1200 data sets; the inset detail shows BR's between 10 and 100.

BRs of >1000.) Within each of these categories, we picked the two data sets for which the difference between the BR calculated by DCUHRE and by MC MISER was the largest; in each such case, MC VEGAS deviated from the fine grid by an even greater amount. We then successively increased the granularity of the fine grid until the resulting BR stabilized. This required increasing the number of fine-grid calculations to a total of 510 million points for each data set.

Figure 4 compares the BRs from the three NI methods with the fine grid, and successive refinements of the fine grid, for these eight data sets. The results suggest that the original fine grid was itself providing an overestimate of the BR, and that as we increase granularity to the point of stabilization, it tends to converge approximately to the results returned by both MC MISER and DCUHRE. By contrast, MC VEGAS severely underestimates BRs, especially in the medium to high range. We believe that the reason for the poor performance of MC VEGAS is its inability to appropriately handle partial integrands when their projections onto the axes are not typical of the general behavior of the integral. Overall, in view of the results based on refinements of the fine grid, we conclude that MC VEGAS is not returning accurate estimates of the BR, and we exclude it from further consideration.

This leaves us with MC MISER and DCUHRE, which return values close to one another and also to the finest grid, but without either one consistently being closest to the finest grid. While in the absence of an analytic solution to the integral, it is impossible to know in any given data set which of the three methods is in fact the most accurate, it is also clear that either MC MISER or DCUHRE will return approximations to the BR that will be highly accurate in general, and certainly accurate enough for our purposes. Thus, viewed solely in terms of accuracy, either method seems appropriate for our application.

However, there are reasons to prefer DCUHRE apart from the question of accuracy. First, as pointed out above, MC MISER shows some inconsistency with regard to the shape of the BR viewed as a function of θ, as seen in Figure 2; thus, even though the BR's are fairly close to those from the fine grid and DCUHRE, inferences that depend upon the shape of the curve (such as the PPL itself) will be subject to an additional layer of approximation. This behavior might change with a different random number generating method, for

instance, we were to use quasi-random numbers. On the other hand, given that DCUHRE is just as accurate while being non-stochastic and therefore independent of the random number generating method, it seems safer to rely on DCUHRE.

But an even more compelling argument in favor of DCUHRE is related to efficiency, rather than accuracy. DCUHRE is able to make use of information regarding the shape of the integrand as it performs the calculation, continuing to evaluate additional points only as long as these additional points are likely to appreciably change the final result. For MC MISER, the number of function evaluations needs to be determined before its application. In performing genome-wide linkage analysis, we know in advance that almost all parts of the genome will not show evidence of linkage, in which case, the integrands will tend to be quite flat across the parameter space. DCUHRE handles this situation implicitly and efficiently, requiring on average only about $10*273 = 2{,}730$ calculations per position when there is no linkage. (Note that this represents a substantial improvement in efficiency over our original standard grid as well.) But MC MISER will require $10*4500 = 45{,}000$ calculations even at these positions. On a genome-wide basis (typically 3774 positions for a genome scan), this can amount to a savings of around 159 million calculations relying on DCUHRE rather than MC MISER.

Overall, we conclude that DCUHRE produces the best results, in terms of both accuracy and efficiency, among all choices we have considered. However, one issue remains to be discussed, and this is the relationship between DCUHRE and the original standard grid. As mentioned previously, the standard grid was arrived at heuristically on the basis of extensive simulations and applications to real data. It was specifically selected for its ability to clearly distinguish "signal" from "noise," that is, to return large values at "linked" loci and small values ($\mathrm{PPL} < 2\%$) at "unlinked" loci. As illustrated in Figure 2, however, DCUHRE systematically returns BRs that are substantially smaller than those from the standard grid in the middle to high range, and also, somewhat larger than those from the standard grid in the small range. (This pattern remains consistent across all 1200 replicates.) This means that in application to real data, PPLs computed using the standard grid will tend to be larger when there is linkage, relative to DCUHRE, and smaller when there is no linkage. Thus, DCUHRE may very well be more accurate as an NI method, but the standard grid appears to be *more useful* as a method for finding genes.

In fact, we understand quite a bit about why this happens. The salient relevant feature of the underlying likelihoods is that they "like," for genetic reasons, to maximize around the edges of the parameter space. In particular, at the value $f_{dd} = 0$, it is possible to get very large spikes in the BR at a linked marker, as well as very large depressions of the BR at an unlinked marker. The standard grid assigned this particular point a large "weight" of 0.24, because 65 out of 275 combinations satisfying the ordering constraints on the penetrances have $f_{dd} = 0$. DCUHRE of course assigns a weight of close to 0 to this particular value, because it is performing genuine (continuous) numerical integration. (Other features of the standard grid, in particular, the uneven grid for the gf, also contribute to its behavior in a similar manner, however, to a lesser extent.)

Thus in choosing between DCHURE and the standard grid, we have in addition to considerations of accuracy and efficiency, the consideration of utility. A more statistical way to frame the situation is to think of the "weights" as forming a prior probability distribution over the parameter space. Because the standard grid utilizes an underlying discrete uniform prior distribution, we were able to easily up- or down-weight points to achieve a useful function of the BR itself; but DCUHRE implicitly utilizes a continuous uniform prior distribution, which as we have seen, for genetic reasons tends to be less effective. On the grounds of elegance and efficiency, there is no question that DCUHRE is preferable to the standard grid (and probably, less apt to lead to errors), and for this reason we select DCUHRE as our new "gold standard" going forward. However, we have also implemented a simple way to "tweak" DCUHRE's calculation in order to more closely mimic the practical advantages of the standard grid. We return to this topic below.

3.2. Optimizing DCUHRE for calculation of the PPL

In this section, we consider three refinements to DCUHRE aimed at optimizing its efficiency and utility in application to linkage analysis. First, we consider integration over θ and/or D' to convert the BR into the PPL; we then introduce an additional "layer" of integration for efficient handling of the parameter α; finally, we develop an error analysis approach for ensuring sufficient accuracy in real applications.

3.2.1. Computation of the PPL from the BR. Whereas calculation of the BR is a difficult task, integration over θ is a simple one. BR's always take one of three forms: monotonically increasing,

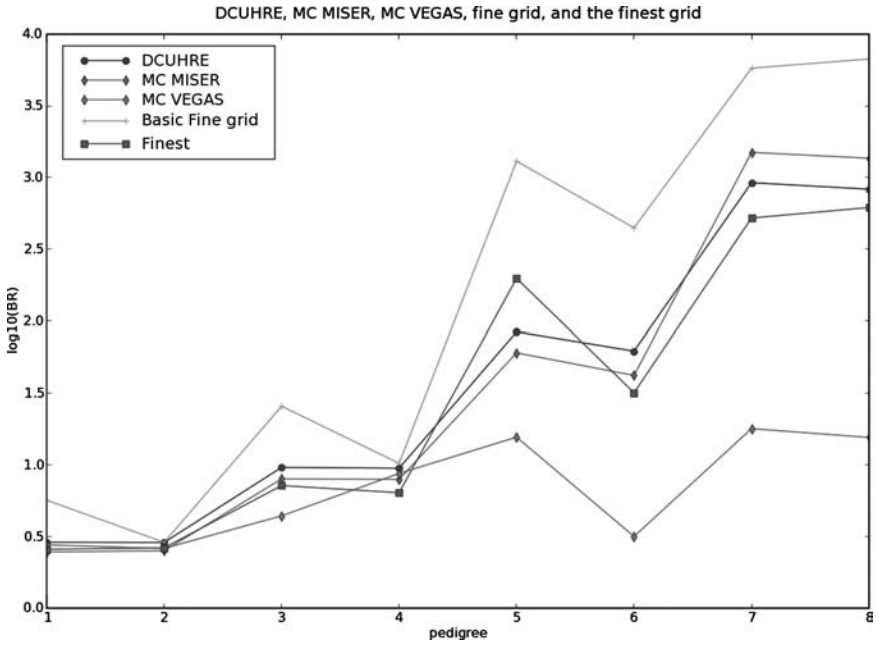

FIG. 4. Comparison of DCUHRE, MC MISER and MC VEGAS on two pedigree data sets from each of four categories, compared with the fine grid (330 million points) and the finest grid (510 million points).

monotonically decreasing, or concave up. The trapezoidal integration method we have applied to the fixed-grid scheme will not work well, especially when integrands are concave up or down in a monotonically increasing or decreasing function, because we cannot expect any offset of errors (that is, errors of all trapezoids are added to increase the total error). Even for the LD_PPL, we observe the same very simple behavior for $BR(\theta,D')$. Thus, a simple one-dimensional quadrature rule for the PPL, or a two-dimensional DCUHRE rule such as a 13-rule for the LD_PPL, should work well here. In order to use a weighting scheme on the BR's (that is, a non-uniform probability distribution) as described previously, we can simply use two sets of Gauss-Legendre points or DCUHRE rule points. The one-dimensional quadrature rule we use is two sets of five-point Gauss-Legendre quadrature rules, which guarantees exact calculations for polynomials up to degree 9. Figure 5 shows the grid scheme we have used in the past and the two-dimensional DCUHRE points. The grid scheme has $21*51 = 1071$ points, and two sets of DCUHRE 13-rules on the two-dimensional space have $65 + 65 = 130$ points. Therefore, about five and eight times fewer BRs are used to calculate the PPL and the LD_PPL, respectively.

We compared these two schemes for calculating PPLs and LD_PPLs using the 54 LD data sets described above. Table 1 shows the results. As can be seen, results based on the two-dimensional fully symmetric rules are very close to those based on the fixed grid points, while requiring a far smaller number of calculations. We know, moreover, that the two-dimensional integration with DCUHRE is actually more accurate.

3.2.2. A three-layer numerical integration approach with DCUHRE. There is still one argument in favor of using a fixed-grid system over DCUHRE to calculate the BRs. Function evaluation for the given variables $(gf, f_{DD}, f_{Dd}, f_{dd}, \alpha)$ requires polynomial evaluations with $(gf, f_{DD}, f_{Dd}, f_{dd})$ in (3), followed by evaluations over α. Since the polynomial evaluations are the most time-consuming aspect of the calculation, we can gain efficiency by reusing evaluations. In particular, when evaluating the polynomials over different values of α, we would prefer not to re-evaluate BRs in the other parameters that values already been computed. The fixed-grid systems can reuse all polynomial evaluations for α in (9). This means that there are only $(\#\ gf)\times(\#\ penetrance)$ combination polynomial evaluations required for the calculation of $BR(\theta)$. However, DCUHRE can achieve only 20% improvement by reusing polynomial evaluations when applied to five-dimensional sub-regions.

To address this issue, we present three approaches to reusing polynomial evaluations. In the first, α stays in the sub-region adaptive scheme and polynomial evaluations of the four other variables are stored for reuse within each sub-region, for the 20% improvement in speed. The other two approaches leave α out of the

DCUHRE scheme, that is, we run DCUHRE on four-dimensional regions (273 points compared to 153 points per a four-dimensional sub-region). This is possible because of the distinctive way in which α enters the likelihood as in (3). Then at each function evaluation, a one-dimensional integration is carried out over α. Here we can either take average function values over evenly spaced points, as in the standard grid, or apply a quadrature rule like Gauss-Legendre. Thus the calculation of the BR is now divided into two separate integrations: the one-dimensional integration corresponds to the inner-most layer of our three-layer scheme, and any function evaluation for the BR invokes numerical integration over α. We have found in practice that either of these last two approaches is preferable to the first approach, because they achieve faster computation time while returning virtually identical results. For the proposed NI method, we use five-point Gauss-Legendre rule because of less computational time and rigorous accuracy.

3.2.3. Error analysis. Here we consider the use of error analysis to help ensure the accuracy of the approximation in each data set. By setting a target error tolerance for the PPL itself, denoted tol_{PPL}, we can calculate the corresponding error tolerance for the integrals from which the PPL is computed, tol_I. With some algebra, we have

$$tol_I < tol_{PPL} \frac{(I_{LE} + (1 - P(L))/P(L))^2}{(1 - P(L))/P(L)}. \tag{10}$$

The prior probability of linkage, $P(L)$ is set to 0.02, as discussed above. Even with small I_{LE} close to 1.0, tol_I becomes approximately 50 * tol_{PPL}. That is, whatever our error tolerance for the PPL, we have 50 times higher tolerance in calculating the integral. When I becomes big, tol_I dramatically increases in (10). This explains why the differences in results across different integration methods decrease for higher PPLs. Figure 2 shows an example (second panel) for which the integrals show very large differences across methods, but the PPLs are very close across methods (0.9898, 0.9655, and 0.9693) for the standard grid, fine grid, and DCUHRE, respectively.

However, while tol_I is easily calculated from tol_{PPL}, ensuring whether or not we are obtaining tol_I is not so simple. For example, error analysis for fully symmetric rules can be developed in terms of derivatives of the integrands. But these are for the most part useless in this context because of the complexities of our integrands and because this complexity varies from one data set to another. This feature presents obstacles to performing theory-based error analysis. Thus the typical error analysis involves computing approximations with successively larger amounts of computing resources, taking the stabilization of the approximations as evidence of convergence to the correct value. While other stopping rules are sometimes formalized, they are based on this same underlying idea. In future work, we plan to develop a stopping criterion to ensure the target error tolerance for the PPL.

3.3. Some practical adaptations and application to real data

As discussed above, DCUHRE tends to return smaller values of the BR when there is linkage, and larger values when there is no linkage, compared to our original standard grid. Thus we initially thought that following full implementation of DCUHRE, we would need to return to the development of appropriate prior distributions over the parameter space. However, there is a much simpler way to induce DCUHRE to mimic the behavior obtained with our original (discrete) prior, and that is simply to "boost" the signal by exponentiation of the BR estimate in the course of conversion onto the PPL scale. After some experimentation, we determined that exponentiation of the BR to the 1.1 power largely restores the original behavior; reforming larger BRs of >1, and smaller BRs of <1, which translates into larger PPLs at linked loci (on average) and smaller ones at unlinked loci. At the same time this ad hoc adjustment appears to retain good differentiation among signals in the small to high range, as did the original standard grid. Obviously, exponentiating to a high power would "boost" signals in both directions even more dramatically. But it would also leave us with a sampling distribution of PPLs in which they were all either (approximately) 0 or 1, without the ability to distinguish loci with moderate evidence for or against linkage from those with very strong evidence. By using just a small boost, we maintain a large and relatively uniform distribution of BRs across the full sampling range in our simulated data.

Moreover, application of this "adjustment" turns out to restore another key property of PPLs. We had previously described a method for maintaining the same scale for two-point linkage analysis (as described in detail above) and multi-point linkage analysis (Logue and Vieland, 2005b). In the latter case, multiple

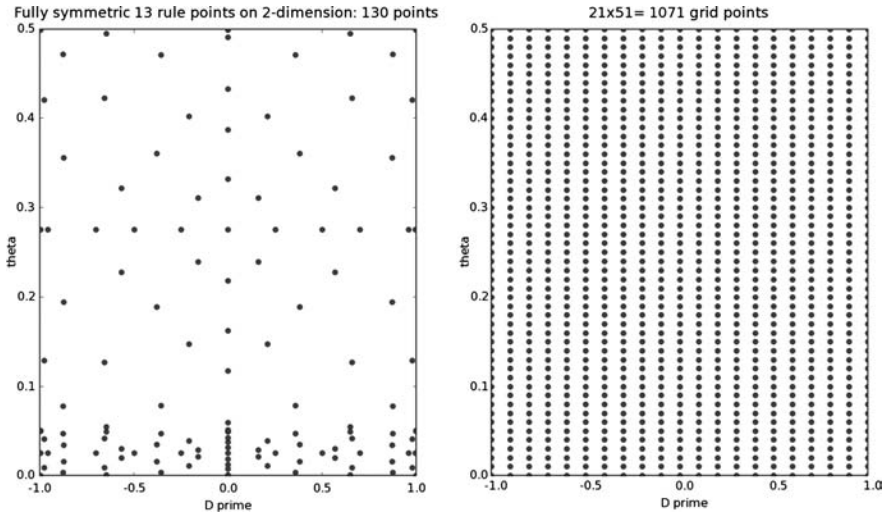

FIG. 5. Points used for DCUHRE scheme and original grid scheme for $D'*\theta$.

markers are used simultaneously to obtain the posterior probability of a trait gene at a specific genomic location: θ is no longer a parameter of the model (because the trait location is fixed against the multi-marker map), and we obtain the multipoint PPL by imputation from the multipoint BR (Logue, 2005b). As shown in the Appendix, the adjusted PPL allows us to retain the original multipoint imputation formula, further confirming the close relationship between the PPL calculated using DCUHRE and the original form of the PPL.

Finally, we show an application of our implementation of DCUHRE to a real data set. Figure 6 shows both DCUHRE and the adjusted form in comparison with the original standard grid, in application to the AITD data set described above. As can be seen, across virtually all of the genome, all three methods agree closely and yield the same overall picture. However, at a few spots DCUHRE returns appreciably smaller PPLs. Not coincidentally, these are all positions at which both the standard grid and DCUHRE are returning non-negligible evidence in favor of linkage (Vieland et al. 2008). Thus, while we would conclude in either case that these were the most likely positions for AITD genes, the signal to noise ratio would be higher using the standard grid. Using the adjusted form of DCUHRE, by contrast, these positions show slightly higher PPLs, while maintaining very similar results elsewhere in the genome. While it is not possible at this time to say with certainty that either method has correctly determined the positions of AITD genes, the results have considerable face validity and some corroborating evidence regarding AITD itself (Vieland et al., 2008). But in any event, we believe that whether the evidence is pointing to true locations, or whether possibly the evidence is misleading in this particular case (as can happen with any statistical analysis), the adjusted form of the DCUHRE-calculated PPL is correctly maintaining the characteristics of the original PPL that have made it so useful in practice, while introducing a level of mathematical rigor hitherto lacking in our approach to NI, and yielding orders of magnitude improvements in computational efficiency.

4. DISCUSSION

By comparing several approaches to NI in application to genetic data, we have found DCUHRE to be the most accurate, indeed, in some applications more accurate even than calculation based on an extremely fine

Table 1. Average PPL and LD_PPL Calculated Using the Original Fixed Grid (PPL(G), LD_PPL(G)) or the DCUHRE Points as Shown in Figure 5 (PPL(D), LD_PPL(D)), Based on 54 Simulated Replicates

Data	*PPL(G)*	*PPL(D)*	*Diff*	*LD_PPL(G)*	*LD_PPL(D)*	*Diff*
L52	0.020154	0.020154	1.7E-6	0.02164	0.021706	1.67E-4
H2	0.021398	0.021391	7.5E-6	0.75465	0.74675	0.0079

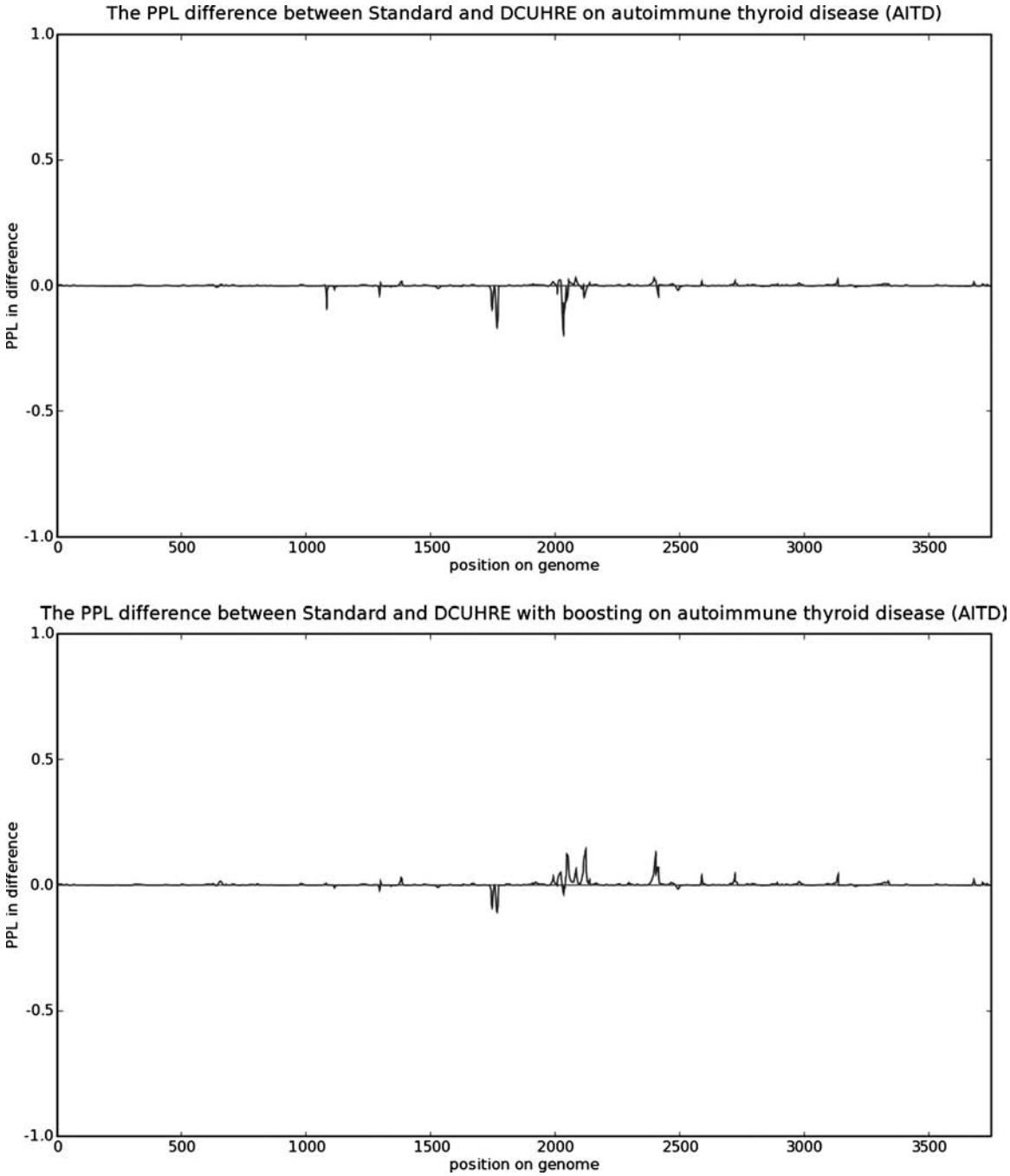

FIG. 6. DCUHRE and the "boosted" form in comparison with the original standard grid on autoimmune thyroid disease (AITD). PPLs by standard grid are subtracted from those by each form of DCUHRE. Positions are in centi-Morgan (cM) and accumulated across the whole genome.

fixed grid scheme; and of course, it is far more efficient than such a scheme. By implementing some application-specific optimizations, we have been able to further increase our efficiency while maintaining high accuracy in the calculation of the PPL. Our new method is about 31 thousand times faster than a fixed grid scheme for equivalent or better accuracy. In practice, this makes genome-wide PPL calculations quite feasible. For example, about 12 hours are spent to complete a genome-wide linkage or linkage disequilibrium scan that previously took more than 2 years using the brute force approach.

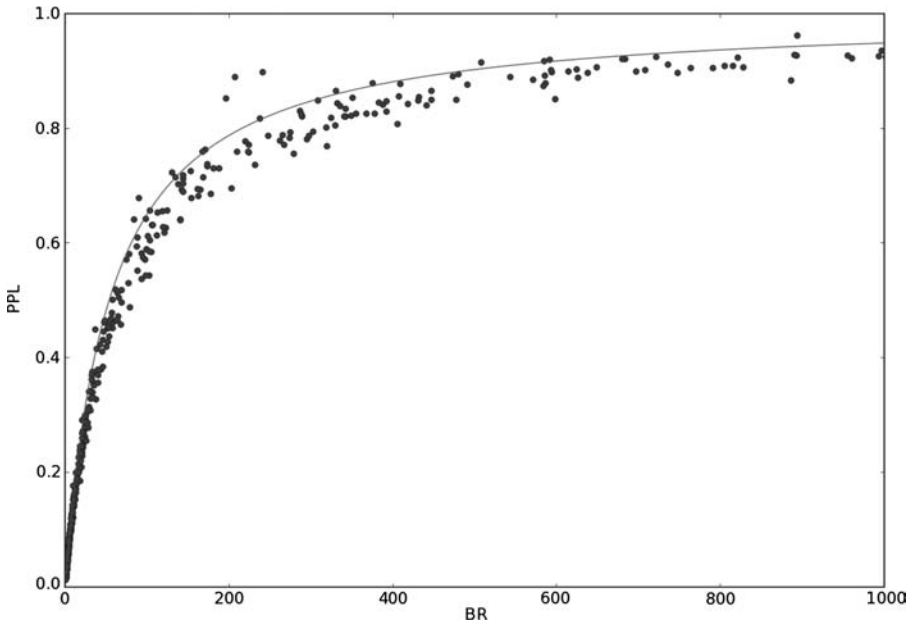

FIG. 7. Two-point PPL's as a function observed BR, both calculated using DCUHRE, compared to original imputed multipoint PPL (shown by curved line), for simulated data sets.

While the proposed three-layer scheme presented here shows outstanding results in both speed and accuracy, some additional fine-tuning of our algorithm remains in order to ensure high precision in all cases. In the majority of data sets considered, the BR calculation achieved our stipulated error tolerance of 1% using far less than the maximum number of DUCHRE function calls (10,000). In cases in which the calculation terminated at 10,000 calls without satisfying this error criterion, the estimated error was usually less than 2%. However, in a small number of cases at the time of termination the error estimate still exceeded 10%. This happened only when the BR's were quite large. Ideally, we would let DCUHRE run as long as necessary to bring the error under the tolerance level, but it is not yet clear whether that will be feasible (that is, how long DCUHRE would continue to run in these cases) or what gain in accuracy would be achieved in the final calculation of the PPL (that is, whether it would be worth the additional computational costs). We plan to pursue a more rigorous approach to error analysis in order to address this issue. In any case, however, our current implementation of DUCHRE appears to be both more accurate and considerably faster than our earlier fixed-grid approach.

Ongoing research is focusing on some extensions to the DCUHRE application. For example, for various genetic reasons, in addition to the PPL, it is frequently also of interest to maximize the HLOD score over all genetic parameters. We would expect DCUHRE to do well at identifying the maximum because in general the region around the maximum will contribute substantially to the integral, therefore be well-covered by DCUHRE's adaptive scheme. In preliminary analyses we have verified its ability correctly detect the maximum HLOD, but we will continue to investigate this feature.

Another subject of ongoing research is extending the current treatment of LD, which assumed just two variants (alleles) at a single genetic marker. However, it is desirable to be able to compute LD_PPL's for multi-allelic markers as well. This entails a vector of D' values rather than the single scalar D' (Yang et al., 2005), with the dimension of the integration increasing proportionally to the number of alleles. This requires as many BR calculations as the number of combinations of θ and D' vector. While we can still use the DCUHRE points for the outermost layer of integration, as the dimensionality becomes high (say, more than 15), we may need either to consider non-deterministic options or to apply DCUHRE to the (θ, D') space for these higher dimensions. The disadvantage of this would be to lose the ability to perform sequential updating in the current form, which requires the use of identical grids across different data sets. For the time being, some form of interpolation might prove useful here.

A further extension of the current PPL framework includes modeling of quantitative traits (QT), for which variables (means, variances) can have infinite range (Bartlett et al., 2007). In establishing a fixed-grid

method, we first verified that accuracy could be achieved while restricting all variables to finite ranges, however in moving to DCUHRE for QT applications, this issue will need to be revisited.

5. CONCLUSION

The PPL is not alone among statistics used in biomedical research in requiring numerical integration, but it does present some unique features against which any proposed integration method must be evaluated. The primary obstacle to implementing the PPL is not only the high burden of computational time, but also the task of ensuring accuracy of the results. Here we analyzed simulated data sets to design and evaluate a three-layer integration strategy, where the main integration method relies on a sub-region adaptive method, DCUHRE. The newly designed method was compared with an empirically derived fixed-grid scheme as well as other alternative methods. The inner-most layer, which is the integration over one variable, α, aims to reuse polynomial evaluations to the maximum extent. The middle layer deals with integrands in the remaining trait parameters. This step focuses on putting more points where the integrands change rapidly, leading not only to fewer number of required function evaluations but also to great improvements in accuracy. These two layers calculate BRs for the given points (θ,D') or just θ. All BRs generated at this stage are stored for future use in sequential updating. The outermost layer calculates BRs at given fully symmetric rule points, and finishes the integration over the localization parameters θ or (θ,D').

We have shown in this article how the proposed method is specifically adapted to our application and how effective it is in providing both faster and more accurate calculations of the PPL.

6. APPENDIX: IMPUTATION FORMULA FOR MULTI-POINT PPL

While somewhat beyond the scope of this article, it is important for practical reasons that the DCUHRE method be adapted to both two-point (one marker at a time) and multipoint (multiple markers at a time) linkage analyses. Previously, we had pointed out drawbacks to alternative Bayesian formulations of multipoint linkage statistics, and established the utility of a particular imputation formula. In brief, the imputation formula works with the BR at each genomic location over which the BR is computed, and directly imputes the two-point PPL that would be obtained if that BR were obtained at $\theta = 0$ at a fully informative marker. (For details and justification, see Logue and Vieland [2005b], whose method ensures that the two-point and multipoint versions of the PPL retain the same scale and avoids the smoothing artifacts of moving-window approaches to multipoint calculations.) The originally derived imputation formula is given below, for calculation of the PPL from the BR at position t_0.

$$PPL_I(t_0) = \frac{(BR(t_0))^2}{-5.77 + 54BR(t_0) + (BR(t_0))^2}.$$

In order to ensure that this imputation formula remains appropriate for our adjusted version of DCUHRE, we applied it to the 1200 replicates described in the main text. Figure 7 shows a scatter plot of two-point PPLs as a function of the observed BRs, both calculated based on this version of DCUHRE, as well as the PPL_I obtained using the original formula, for BR of <1000 as in Logue and Vieland (2005b). The object here is to have the PPL_I curve "fit" the predicted two-point PPL from the single BR value as closely as possible. As can be seen, the imputation formula provides an excellent fit to the data, giving a slight additional upward boost when the evidence supports linkage, but not when the evidence supports the absence of linkage. This result is essentially identical to what was originally observed (Logue and Vieland, 2005b). We therefore conclude that use of the original formula remains valid under the new approach to NI.

ACKNOWLEDGMENTS

We would like to thank Dr. Yaron Tomer for permission to use the AITD data set. This research was funded in part by NIH grants.

DISCLOSURE STATEMENT

No competing financial interests exist.

REFERENCES

Bartlett, C,W., and Vieland, V.J. 2007. Accumulating quantitative trait linkage evidence across multiple datasets using the posterior probability of linkage. *Genet. Epidemiol.* 31, 91–102.

Bartlett, C.W., and Vieland, V.J. 2005. Two novel quantitative trait linkage analysis statistic based on the posterior probability of linkage: application to the COGA families. *BMC Genet*. 6, Suppl 1, S121.

Bartlett, C.W., Flax, J.F., Logue, M.W., et al. 2004. Examination of a potential overlap in autism and language loci on chromosomes 2,7, and 13 in two independent samples ascertained for specific language impairment. *Hum. Hered.* 57, 10–20.

Bartlett, C.W., Goedken, R.J., and Vieland, V.J. 2005. Effects of updating linkage evidence across subjects of data: reanalysis of autism genetic resource exchanging data set. *Am. J. Hum. Genet.* 76, 688–695.

Berntsen, J., Espelid, T., and Genz, A. 1988. A test of ADMINT. *Reports in informatics 31.* Department of Informatics, University of Bergen, Bergen, Norway.

Berntsen, J., Espelid, T., and Genz, A. 1991. An adaptive multidimensional integration routine for a vector of integrals. *ACM Trans. Math. Softw.* 17, 452–456.

Elston, R.C., and Lange, K. 1975. The prior probability of autosomal linkage. *Ann. Hum. Genet.* 38, 341–350.

Elston, R.C., and Stewart, J. 1971. A general model for the genetic analysis of pedigree data. *Hum. Hered.* 21, 523–542.

Evans, M., and Swartz, T. 2000. *Approximating Integrals via Monte Carlo and Deterministic Methods.* Oxford University Press, New York.

Genz, A., and Malik, A. 1983. An imbedded family of fully symmetric numerical integration rules. *SIAM J. Numer. Anal.* 20, 580–587.

Govil, M., and Vieland, V.J. 2008. Practical considerations for dividing data into subsets prior to PPL analysis. *Hum. Hered.* 66, 223–237.

Huang, J., and Vieland, V.J. 2001. Comparison of ''model-free'' and ''model-based'' linkage statistics in the presence of locus heterogeneity: single data set and multiple data set applications. *Hum. Hered.* 51, 217–225.

Lepage, G.P. 1978. A new algorithm for adaptive multidimensional integration. *J. Comp. Phys.* 27, 192–203.

Logue, M., and Vieland, V.J. 2005b. A new method for computing the multipoint posterior probability of linkage. *Hum. Hered.* 57, 90–99.

Logue, M., and Vieland, V.J. 2005a. The incorporation of prior genomic information does not necessarily improve the performance of Bayesian linkage methods: an example involving sex-specific recombination and the two-point PPL. *Hum. Hered.* 60, 196–205.

Logue, M., Vieland, V.J., Goedken, R.J., et al. 2003. Bayesian analysis of a previously published genome screen for panic disorder reveals new and compelling evidence for linkage to chromosome 7. *Am. J. Med. Gen. Neuropsych. Genet.* 121B, 99.

Logue, M.W., Brzustowicz, L.M., Bassett, A.S., et al. 2006. A posterior probability of linkage (PPL) based re-analysis of schizophrenia data yields evidence of linkage to chromosome 1 and 17. *Hum. Hered.* 62, 47–54.

Ott, J. 1999. *Analysis of Human Genetic Linkage*, rev. ed. The Johns Hopkins University Press, Baltimore.

Park, J., Cremer, J., Segre, A., et al. 2006. Visual exploration of genetic likelihood space. *Proc. ACM* SAC, 1335–1340.

Press, W.H., and Farrar, G.R. 1990. Recursive stratified sampling for multidimensional Monte Carlo integration. *Comput. Physics* 4, 190–195.

Smith, C.A.B. 1959. Some comments on the statistical methods used in linkage investigations. *Am. J. Hum. Genet.* 39, 423–426.

Tomer, Y., Ban, Y., Concepcion, E., et al. 2003. Common and unique susceptibility loci in Graves and Hashimoto diseases: results of whole-genome screening in a data set of 102 multiplex families. *Am. J. Hum. Genet.* 73, 736–747.

Vieland, V.J. 1998. Bayesian linkage analysis, or: how I learned to stop worrying and love the posterior probability of linkage. *Am. J. Hum. Genet.* 63, 947–954.

Vieland, V.J. 2006. Thermometers: something for statistical geneticists to think about. *Hum. Hered.* 61, 144–156.

Vieland, V.J., Huang, Y., Bartlett, C., et al. 2008. A multilocus model of the genetic architecture of autoimmune thyroid disorder, with clinical implications. *Am. J. Hum. Genet.* 82, 1349–1356.

Vieland, V.J., Wang, K., and Huang, J. 2001. Power to detect linkage based on multiple sets of data in the presence of locus heterogeneity: comparative evaluation of model-based linkage methods for affected sib pair data. *Hum. Hered.* 51, 199–208.

Wang, H., Segre, A.M., Huang, Y., et al. 2007a. Fast computation of human genetic linkage. *Proc. IEEE 7th Int. Conf. BIBE*, 857–863.

Wang, H., Segre, A.M., Huang, Y., et al. 2007b. Rapid computation of large numbers of LOD scores in linkage analysis through polynomial expression of genetic likelihoods. *Proc. Int. Conf. BIBM*, 197–204.

Wang, K., Huang, J., Logue, M., et al. 2001. Combined multipoint analysis of multiple asthma datasets based on the posterior probability of linkage. *Genet. Epidemiol.* 21, Suppl 1, S73–S78.

Yang, X., Huang, J., Logue, M.W., et al. 2005. The posterior probability of linkage allowing for linkage disequilibrium and a new estimate of disequilibrium between a trait and a marker. *Hum. Hered.* 59, 210–219.

Address reprint requests to:
Dr. Sang-Cheol Seok
Battelle Center for Mathematical Medicine
Research Institute at Nationwide Children's Hospital
700 Children's Drive
Columbus, OH 43205

E-mail: Sang-Cheol.Seok@nationwidechildrens.org