

COWLES FOUNDATION FOR RESEARCH IN ECONOMICS
AT YALE UNIVERSITY

Post Office Box 208281
New Haven, Connecticut 06520-8281

COWLES FOUNDATION DISCUSSION PAPER NO. 1021R

NOTE: Cowles Foundation Discussion Papers are preliminary materials circulated to stimulate discussion and critical comment. Requests for single copies of a Paper will be filled by the Cowles Foundation within the limits of the supply. References in publications to Discussion Papers (other than acknowledgment that a writer had access to such unpublished material) should be cleared with the author to protect the tentative character of these papers.

Simulation of Multivariate Normal Rectangle

Probabilities and their Derivatives:

Theoretical and Computational Results

by

Vassilis Hajivassiliou, Daniel McFadden

and Paul Ruud

October 1994

Simulation of Multivariate Normal Rectangle Probabilities and their Derivatives: Theoretical and Computational Results¹

by

Vassilis Hajivassiliou*
Daniel McFadden**
Paul Ruud***

July 1992, Revised October 1994

ABSTRACT

An extensive literature in econometrics and in numerical analysis has considered the problem of evaluating the multiple integral $P(\mathbf{B}; \mu, \Omega) = \int_a^b n(v - \mu, \Omega) dv \equiv \mathbf{E}_V \mathbf{1}(V \in \mathbf{B})$, where V is a m -dimensional normal vector with mean μ , covariance matrix Ω , and density $n(v - \mu, \Omega)$, and $\mathbf{1}(V \in \mathbf{B})$ is an indicator for the event $\mathbf{B} = \{V \mid a < V < b\}$. A leading case of such an integral is the negative orthant probability, where $\mathbf{B} = \{v \mid v < 0\}$. The problem is computationally difficult except in very special cases. The *multinomial probit* (MNP) model used in econometrics and biometrics has cell probabilities that are negative orthant probabilities, with μ and Ω depending on unknown parameters (and, in general, on covariates). Estimation of this model requires, for each trial parameter vector and each observation in a sample, evaluation of $P(\mathbf{B}; \mu, \Omega)$ and of its derivatives with respect to μ and Ω . This paper surveys Monte Carlo techniques that have been developed for approximations of $P(\mathbf{B}; \mu, \Omega)$, and its linear and logarithmic derivatives, that limit computation while possessing properties that facilitate their use in iterative calculations for statistical inference: the Crude Frequency Simulator (CFS), Normal Importance Sampling (NIS), a Kernel-Smoothed Frequency Simulator (KFS), Stern's Decomposition Simulator (SDS), the Geweke-Hajivassiliou-Keane Simulator (GHK), a Parabolic Cylinder Function Simulator (PCF), Deák's Chi-squared Simulator (DCS), an Acceptance/Rejection Simulator (ARS), the Gibbs Sampler Simulator (GSS), a Sequentially Unbiased Simulator (SUS), and an Approximately Unbiased Simulator (AUS). We also discuss Gauss and FORTRAN implementations of these algorithms and present our computational experience with them. We find that GHK is overall the most reliable method.

JEL Classification: 210

KEYWORDS: Simulation Estimation, Monte-Carlo Integration, Discrete Choice Models, Multinomial Probit Models, Importance Sampling, Acceptance-Rejection, Gibbs Resampling

* Cowles Foundation, Yale University

** Department of Economics, University of California at Berkeley

*** Department of Economics, University of California at Berkeley

¹We have benefitted from interchanges with Chunrong Ai, Michael Keane, Taka Morikawa, Dan Nelson, Anne Royalty, and Steve Stern. We thank two anonymous referees for very constructive comments and suggestions, Carmela Quintos for expert assistance with Gauss, Yoosoon Chang with FORTRAN, and Phil Molner with running the Monte Carlo tests. Hajivassiliou gratefully acknowledges financial support from the National Science Foundation through grants SES-9000200 and SES-9211913.

Simulation of Multivariate Normal Rectangle Probabilities and their Derivatives: Theoretical and Computational Results²

by

Vassilis Hajivassiliou
Daniel McFadden
Paul Ruud

1 Introduction

An extensive literature in econometrics and in numerical analysis³ has considered the problem of evaluating the multiple integral

$$(1) \quad P \equiv P(\mathbf{B}; \mu, \Omega) = \int_a^b n(v - \mu, \Omega) dv \equiv \mathbf{E}_{\mathbf{V}} \mathbf{1}(V \in \mathbf{B}),$$

where V is a m -dimensional normal random vector with mean μ , covariance matrix Ω , and density $n(v - \mu, \Omega)$, and $\mathbf{1}(V \in \mathbf{B})$ is an indicator for the event $\mathbf{B} = \{V \mid a < V < b\}$. A leading case of such an integral is the negative orthant probability, where $\mathbf{B} = \{v \mid v < 0\}$.⁴ The problem is computationally difficult unless the dimension of the integral is less than four or the covariance matrix Ω has a special structure, such as a factorial structure with a small number of factors.

The *multinomial probit* (MNP) model used in econometrics and biometrics has cell probabilities that are negative orthant probabilities, with μ and Ω depending on unknown parameters and, in general, on covariates.⁵ Estimation of this model requires, for each trial parameter vector and each observation in a sample, evaluation of (1) and of its linear and logarithmic derivatives with respect to μ and Ω . This paper surveys Monte Carlo techniques that have been developed for approximation of (1) and its derivatives that limit computation while possessing properties that facilitate their use in iterative calculations for statistical inference. Section 2 presents the simulation methods we consider. An Appendix discusses GAUSS and FORTRAN algorithms implementing these methods. Section 3 describes the test problems we used to investigate the operational properties of the methods and summarizes our computational experience with them. Section 4 concludes the paper.

²This paper collects and extends results obtained by the authors in Hajivassiliou and McFadden (1990), McFadden (1989), McFadden and Ruud (1990), and Ruud (1986, 1991).

³See Clark (1961), Daganzo (1980), Davis and Rabinowitz (1984), Dutt (1973, 1976), Fishman (1973), Hammersley and Handscomb (1964), Horowitz, Sparmann, and Daganzo (1981), Moran (1984), Owen (1956), Rubinstein (1981), Stroud (1971), and Thisted (1988).

⁴Where convenient, we write $P(\mathbf{B}; \mu, \Omega)$ as $P(a, b; \mu, \Omega)$, or when $a = -\infty$, as $P(b; \mu, \Omega)$. Note that $P(b; \mu, \Omega) \equiv P(0; \mu - b, \Omega)$ is the cumulative multivariate normal distribution, also denoted $\mathcal{N}(b; \mu, \Omega)$. This setup covers all cases of interest, since components V_i for which both limits are infinite can be margined out analytically, and components V_i with a_i finite and $b_i = +\infty$ can be converted to the previous case by a reversal of sign.

⁵For example, see McFadden (1984).

2 Simulation Methods

We begin by deriving the linear and logarithmic derivatives of choice probabilities, and then we discuss the simulation approaches we consider in this paper. After introducing some notation in subsection 2.3, we explain the general principles of sequential sampling and acceptance/rejection techniques in subsection 2.4, of unbiased and asymptotically unbiased simulation in subsection 2.5, and of importance sampling in subsection 2.6. We then discuss two variance-reduction methods (antithetic variates in subsection 2.7 and control variates in 2.8). Subsections 2.9 through 2.19 describe the eleven simulation techniques that we analyze in this paper. Table 1 summarizes the methods and the mnemonics we will adopt. We begin with the most direct method, the crude frequency simulator (CFS), which computes the sample frequency of simulated draws from a binomial distribution with probability P . Next we describe a generalization of the CFS which employs the technique of importance sampling, thereby converting the sample frequency to a weighted sample frequency that can have a smaller sampling variance. We call this the normal importance sampling (NIS) simulator. The kernel-smoothed frequency simulator (KFS) generalizes the CFS by replacing the discrete zero-one outcome of the binomial experiment by a continuous outcome on the $[0,1]$ interval. This simulator overcomes the discontinuities in the CFS with respect to the parameters of the underlying normal distribution, but is biased. The fourth through seventh simulators, the Stern-decomposition simulator (SDS), the Geweke-Hajivassiliou-Keane simulator (GHK), the parabolic cylinder function (PCF) simulator, and the Deák Chi-Square Simulator (DCS), are also applications of importance sampling like the NIS and are both smooth and unbiased. These simulators differ according to the importance sampling distribution that they use. All of the simulators, from CFS through DCS, are simulators of P and its derivatives.

A second group of simulators apply specifically to the logarithmic derivatives of P , because these simulators address the additional difficulties posed by the problem of sampling from a truncated normal distribution. The acceptance/rejection simulator (ARS) is another importance sampling technique which additionally filters out draws that fall outside an acceptance region determined by the truncation. The Gibbs sampler simulator (GSS) is an alternative method for sampling from the truncated multivariate normal distribution. The GSS is smooth in the distribution parameters, but the ARS is not. A third approach is taken in the sequentially unbiased simulators (SUS) which construct unbiased simulators of $1/P$. The last method, approximately unbiased simulators (AUS), comprises a family of simulators that are approximately unbiased for $1/P$. The members of this family can be constructed from most of the simulators of P in the first group.

Table 1
Simulators for P , $\nabla_{\mu}P$, $\nabla_{\Omega}P$, $\nabla_{\mu} \log P$, and $\nabla_{\Omega} \log P$

Subsection	Name of Simulator	Mnemonic	Unbiased for P	Unbiased for ∇P	Unbiased for $\nabla \log P$
2.9	Crude Frequency Simulator	CFS	y	n	n
2.10	Normal Importance Sampling Simulator	NIS	y	n	n
2.11	Kernel-Smoothed Frequency Simulator	KFS	y*	n	n
2.12	Stern Decomposition Simulator	SDS	y	n	n
2.13	Geweke-Hajivassiliou-Keane Simulator	GHK	y	n	n
2.14	Parabolic Cylinder Function Simulator	PCF	y	n	n
2.15	Deák Chi-square Simulator	DCS	y	n	n
2.16	Acceptance/Rejection Simulator	ARS	-	-	y
2.17	Gibbs Sampler Simulator	GSS	-	-	y**
2.18	Sequentially Unbiased Simulator	SUS	n	n	n
2.19	Approximately Unbiased Simulator	AUS	n	n	n

* Window parameter must approach 0.

** Number of Gibbs resamplings must approach ∞ .

2.1 Derivatives of Rectangle Probabilities

The derivatives of (1) with respect to μ and Ω can be written:⁶

$$(2) \quad \nabla_{\mu}P(\mathbf{B}; \mu, \Omega) = \Omega^{-1} \int_{-\infty}^{+\infty} \mathbf{1}(v \in \mathbf{B})(v - \mu)n(v - \mu, \Omega)dv \equiv \Omega^{-1} \mathbf{E}_{\mathbf{V}} \mathbf{1}(V \in \mathbf{B})(V - \mu),$$

$$(3) \quad \nabla_{\Omega}P(\mathbf{B}; \mu, \Omega) = (1/2)\Omega^{-1} \int_{-\infty}^{+\infty} \mathbf{1}(v \in \mathbf{B})[(v - \mu)(v - \mu)' - \Omega]n(v - \mu, \Omega)dv \cdot \Omega^{-1} \equiv (1/2)\Omega^{-1} \mathbf{E}_{\mathbf{V}} \mathbf{1}(V \in \mathbf{B})[(V - \mu)(V - \mu)' - \Omega] \cdot \Omega^{-1}.$$

These formulas imply

$$(4) \quad \nabla_{\mu} \log P(\mathbf{B}; \mu, \Omega) \equiv \nabla_{\mu}P(\mathbf{B}; \mu, \Omega)/P(\mathbf{B}; \mu, \Omega) = \Omega^{-1} \mathbf{E}_{\mathbf{V}|\mathbf{B}}(V - \mu),$$

$$\nabla_{\Omega} \log P(\mathbf{B}; \mu, \Omega) \equiv \nabla_{\Omega}P(\mathbf{B}; \mu, \Omega)/P(\mathbf{B}; \mu, \Omega) =$$

$$(5) \quad (1/2)\Omega^{-1} \mathbf{E}_{\mathbf{V}|\mathbf{B}}[(V - \mu)(V - \mu)' - \Omega] \cdot \Omega^{-1},$$

where " $\mathbf{E}_{\mathbf{V}|\mathbf{B}}$ " denotes expectation with respect to the conditional density $n(v - \mu, \Omega, \mathbf{B}) \equiv \mathbf{1}(v \in \mathbf{B})n(v - \mu, \Omega)/P(\mathbf{B}; \mu, \Omega)$. Note that (2) and (3) are partial moments of the density, and (4)

⁶Equation (3) is derived without imposing symmetry on the elements of Ω , and is then simplified by using the symmetry of Ω at the evaluation point. These formulas are due to Ruud (1986), and they can be demonstrated using the matrix derivatives $\partial \log |A| / \partial A = A^{-1}$ and $\partial A^{-1} / \partial A = -A^{-1} \otimes A^{-1}$. If Γ is a lower triangular Choleski factor of Ω , so that $\Omega = \Gamma\Gamma'$, then $\nabla_{\Gamma}P = 2(\nabla_{\Omega}P)\Gamma$.

and (5) are conditional moments. It is useful for statistical applications to develop techniques for approximating (4)–(5) as well as (1)–(3).

Equations (1)–(3) can be written

$$(6) \quad H \equiv H(\mathbf{B}; \mu, \Omega) = \int_{-\infty}^{+\infty} \mathbf{1}(v \in \mathbf{B})h(v)n(v - \mu, \Omega)dv \equiv \mathbf{E}_V \mathbf{1}(V \in \mathbf{B})h(V),$$

and equations (4) and (5) can be written

$$(7) \quad H_C = \mathbf{E}_{V|\mathbf{B}} h(V) \equiv H/P(\mathbf{B}; \mu, \Omega),$$

where $h(v)$ is the polynomial array

$$(8) \quad h(v) = \begin{bmatrix} 1 & (v - \mu)' \Omega^{-1} \\ \Omega^{-1}(v - \mu) & \frac{1}{2}[\Omega^{-1}(v - \mu)(v - \mu)' \Omega^{-1} - \Omega^{-1}] \end{bmatrix}.$$

Then the northwest element of H gives (1), the remainder of the first row gives (2), and the southeast subarray gives (3); the analogous elements of H_C give one, (4), and (5).

2.2 Simulation Procedures

For statistical inference, it is often unnecessary to achieve high numerical accuracy in evaluation of (1)–(5). For example, simulating P by the frequency of the event $\mathbf{1}(v \in \mathbf{B})$ in a number of Monte Carlo draws comparable to sample size will tend to produce statistics in which the variance introduced by simulation is at worst of the same magnitude as the variance due to the observed data. Further, when probabilities appear linearly across observations in an estimation criterion, independent unbiased simulation errors are averaged out. Then, a small, fixed number of draws per probability to be evaluated will be sufficient with increasing sample size to reduce simulation noise at the same rate as noise from the observed data.⁷ This makes it computationally feasible to treat statistical problems that require repeated evaluation of high-dimensional normal rectangle

⁷In outline, suppose $\hat{\theta}$ is an M -estimator that solves

$$0 = N^{1/2} \mathbf{E}_N s(\hat{\theta}, \eta),$$

where s is an approximation (involving Monte Carlo elements η) to a function $\sigma(\theta)$ of P and its derivatives that has expectation zero at the true parameter θ° , and \mathbf{E}_N denotes empirical expectation over an independent sample of size N . Then, one can write

$$0 = N^{1/2} \mathbf{E}_N s(\hat{\theta}, \eta) \equiv N^{1/2} \mathbf{E}_N \sigma(\theta^\circ) + N^{1/2} \mathbf{E}_N [s(\theta^\circ, \eta) - \sigma(\theta^\circ)] + N^{1/2} \mathbf{E}_N [\sigma(\hat{\theta}) - \sigma(\theta^\circ)] \\ + N^{1/2} \mathbf{E}_N [s(\hat{\theta}, \eta) - \sigma(\hat{\theta}) - s(\theta^\circ, \eta) + \sigma(\theta^\circ)].$$

Under standard regularity conditions, the first term is asymptotically normal, reflecting the noise in the observations, and the third term is proportional to $\sqrt{N}(\hat{\theta} - \theta^\circ)$. The last term will be of order $o_p(1)$ for simulators that satisfy a stochastic equicontinuity condition. When s is a smooth function of crude frequency simulators of P , $\nabla_\mu P$, etc., obtained using R Monte Carlo draws, the second term will behave like $\sqrt{N/R}$ times an expression that is asymptotically normal, so that it will be comparable in magnitude to the first term when R and N are proportional. If, in addition, there is any averaging out of simulation noise across observations, the second term may be of order $o_p(1)$ when R and N are proportional, or comparable in magnitude to the first term for fixed R .

probabilities. McFadden (1989), Pakes and Pollard (1989), and McFadden and Ruud (1990) analyze the statistical properties of such estimators, and Hajivassiliou (1993) surveys simulation estimation methods for limited dependent variable models.

The first seven methods we will discuss in this paper, CFS through DCS, simulate H . Methods ARS and GSS simulate H_C by drawing from the conditional distribution of V given $V \in \mathbf{B}$. Method SUS approximates $H_C = H/P$ using independent unbiased simulators of H and $1/P$. Method AUS is similar, but uses a biased simulator of $1/P$ to speed computation. Some versions of AUS require a positive simulator of P . This is guaranteed by NIS, SDS, GHK, PCF, DCS, and by KFS if a positive kernel is used. The number of draws required in ARS is random. The remaining methods will in general use a fixed number of repetitions, which may in statistical applications increase with sample size.

To understand what is perhaps the most intuitive simulation method, write the random vector V as

$$(9) \quad V = \mu + \Gamma\eta,$$

where η is an independent standard normal vector of dimension m and Γ is a lower triangular Choleski factor of Ω , so $\Omega = \Gamma\Gamma'$. A simple approach to approximating (6) is to make repeated Monte Carlo draws for η , use (9) to calculate V for each parameter vector, and then form an empirical analogue of the expectation in (6). Below we call this the *crude frequency simulator* (CFS) of $P(\mathbf{B}; \mu, \Omega)$ and its derivatives. Similarly, a crude frequency simulator for H_C can be formed by rejecting draws of V that do not satisfy the conditioning event $V \in \mathbf{B}$, and then forming an empirical analogue of the conditional expectation in (7) using the accepted draws. The advantages and disadvantages of the frequency simulators are discussed in section 2.9 below.

2.3 General Notation

Before discussing the simulation techniques, it is useful to introduce the following notation. For a vector of indices $(1, \dots, n)$, we use the notation “ $\langle j$ ” to denote the subvector $(1, \dots, j-1)$, and “ $-j$ ” to denote the subvector that excludes component j . Thus, for a matrix Γ , $\Gamma_{j, \langle j}$ denotes a vector containing the first $j-1$ elements of row j , and $\Gamma_{-j, -j}$ denotes the subarray excluding row j and column j . For a vector η , $\eta_{\langle j}$ is the subvector of the first $j-1$ components, and η_{-j} is the subvector excluding component j .

2.4 Sequential Sampling and Acceptance/Rejection Methods

Some simulation techniques, such as the crude frequency simulators for H_C that reject draws not in \mathbf{B} , require sequential sampling. Others, such as the crude frequency simulator for P , may employ sequential sampling for some purposes such as approximating $1/P$. Sequential sampling may involve excessive computation if the yield of accepted points is low, so there is a payoff to techniques that increase yields. Sequential schemes also introduce the possibility of occasional

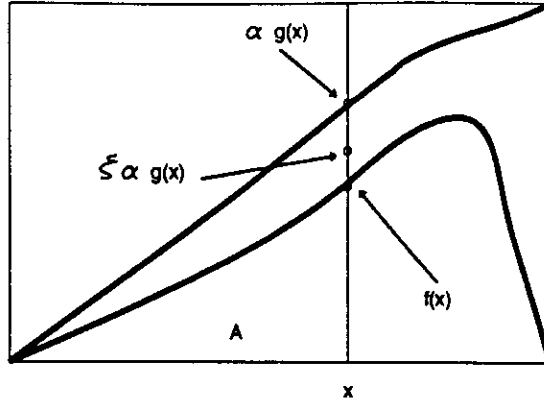


Figure 1: Acceptance/Rejection

lengthy computations. For computer implementation it is desirable to bound computation by building censoring rules into the sampling algorithms and taking account of the effects of this censoring on the statistical properties of the simulators.

Acceptance/Rejection methods provide a mechanism for drawing from a conditional density when practical exact transformations from uniform or standard normal variates are not available. The following result is standard; see Devroye (1986) or Rubinstein (1981).

Lemma 1. *Suppose $f(x)$ is an m -dimensional density, and one wishes to sample from the conditional density $f(\cdot | A)$ given the event $x \in A$. Suppose $g(x)$ is a density from which it is practical to sample, with the property that*

$$\sup_A f(x)/g(x) \leq \alpha < +\infty.$$

Assume that either the support of g is A , or that it is practical to test if $x \in A$; that it is practical to calculate $f(x)$ and $g(x)$; and that it is practical to calculate a bound α . Draw x from g and ζ from a uniform density on $[0,1]$, repeat this process until a pair satisfying $x \in A$ and $f(x) \geq \zeta \alpha g(x)$ is observed, and accept the associated x . Then, the accepted points have density $f(x|A) \equiv f(x)/f(A)$.

Proof: Figure 1 shows schematically that this method yields accepted points with the density $f(x)/f(A)$: One draws points uniformly from the area under αg , and rejects them if they are above f , leaving a uniform distribution of points from the area f : $\text{Prob}(x \& \text{accept}) = g(x) \cdot [f(x)/g(x)\alpha]$, implying $\text{Prob}(x|\text{accept}) = f(x)/f(A)$, as desired. \square

Suppose now that acceptance/rejection is used with a *censoring rule* that if the first r trials do not yield an acceptance, then the last draw from $g(x)$ is accepted unconditionally. This method will be biased, but the bias is bounded at a geometric rate in r :

Lemma 2. *Suppose the assumptions of Lemma 1 hold, with A the support of g . Suppose an acceptance/rejection sampling rule, with censoring after r trials. Let $f^{(r)}(x|A)$ denote the density of x obtained by the censored A/R rule. Then, for any measurable function $h(x)$ such that the expectation $\int h(x)g(x)dx$ exists,*

$$(10) \quad \left| \int h(x)f(x|A)dx - \int h(x)f^{(r)}(x|A)dx \right| \\ \leq \left| \int h(x)[g(x) - f(x)/f(A)]dx \right| \cdot [1 - f(A)/\alpha]^r.$$

Proof: The probability of acceptance on a trial, given x is drawn, is $f(x)/\alpha g(x)$, so that the unconditional probability of acceptance is

$$\int g(x)[f(x)/\alpha g(x)]dx = f(A)/\alpha.$$

Then, the probability of a rejection on the first r trials is $[1 - f(A)/\alpha]^r$, and the probability of x and acceptance on trial i is

$$[1 - f(A)/\alpha]^{i-1} [f(x)/\alpha g(x)]g(x).$$

Then, the probability of x is

$$f^{(r)}(x|A) = \sum_{i=1}^r [1 - f(A)/\alpha]^{i-1} f(x)/\alpha + [1 - f(A)/\alpha]^r g(x) \\ = \frac{1 - [1 - f(A)/\alpha]^r}{1 - [1 - f(A)/\alpha]} \cdot f(x)/\alpha + g(x)[1 - f(A)/\alpha]^r \\ = f(x|A) + [g(x) - f(x)/f(A)] \cdot [1 - f(A)/\alpha]^r.$$

Since $\alpha g(x)/f(A)$ dominates $f(x)/f(A)$, $\int h(x)f(x|A)dx$ exists. Then forming the expectation using the formula above gives the result. \square

2.5 Unbiased and Asymptotically Unbiased Simulators

The crude frequency simulators of H and H_C are all *unbiased* with finite variance, desirable properties that guarantee that when these expressions appear linearly in a condition defining a statistical estimator and the draws are independent across observations, a central limit theorem will operate to attenuate the effect of simulation noise on the estimator. For H , there are unbiased simulators that are smooth in parameters; we have not found smooth unbiased simulators for H_C . It may be advantageous for computation to work with smooth biased simulators, and increase the number of draws in statistical applications as iterative search proceeds or as sample size increases, to eliminate bias in the limit.

An important case where bias is an issue is simulation of $1/P$ in the expressions $(\nabla_\mu P)/P$ and $(\nabla_\Omega P)/P$ for the logarithmic derivatives. Obviously, an unbiased simulator of P such as the

crude frequency simulator for (1) does not yield an unbiased simulator of $1/P$. One technique for achieving an unbiased simulator is based on the observation that $1/P$ is the expectation of the geometric distribution of the number of independent draws R from (9) required to yield $V \in B$; this can be simulated by drawing sequentially from (9) until $V \in B$ is observed. For P small, this method can require a large number of simulations. The following result, due to Ruud, refines the method first by using an upper bound on P , and second by reducing the variance in the simulation of the run length.

Lemma 3. *Assume there is an event A such that*

(i) $E_V \mathbf{1}(V \in A)$ can be computed exactly,

(ii) $B \subseteq A$,

and

(iii) it is convenient to draw from $n(v - \mu, \Omega)$ conditioned on the event A.

Draw sequentially from the conditional distribution of V given A until $V \in B$ is observed, and let R denote the number of draws required. Then, $R/E_V \mathbf{1}(V \in A)$ is an unbiased simulator of $1/P$. The variance in the simulator is reduced by defining

$$(11) \quad R = 1 + Q_1(1 + Q_2(1 + Q_3(1 + \dots))),$$

where each Q_i is an independent unbiased simulator of $E_{V|A} \mathbf{1}(V \notin B)$, initially the Q_i are smooth simulators, and eventually $Q_i = \mathbf{1}(V_i \notin B)$, where the V_i are draws of V given A.

Proof: $ER = Pr(A)/Pr(B) = 1/E_{V|A} \mathbf{1}(V \in B)$; then $ER/E_V \mathbf{1}(V \in A) = 1/P$. The alternative form for R has $ER = 1 + (EQ_1)(1 + (EQ_2)(1 + \dots)) = \sum_{i=0}^{\infty} (1-P)^i = Pr(A)/Pr(B)$, as required.

□

If A can be chosen so that $Pr(A)/Pr(B)$ is not too large, then the expected number of draws required in (11) is not impractically large, even for small P . For example, the event A that $a_i < V_i < b_i$, with i a specific component of V , has the properties

(i) $E_V \mathbf{1}(V \in A) = \Phi((b_i - \mu_i)/\Omega_{ii}^{1/2}) - \Phi((a_i - \mu_i)/\Omega_{ii}^{1/2})$,

(ii) $A \subseteq B$,

and

(iii) it is easy to draw V conditioned on A by first drawing

$$(12) \quad V_i = \Phi^{-1}(\zeta \Phi((b_i - \mu_i)/\Omega_{ii}^{1/2}) + (1 - \zeta) \Phi((a_i - \mu_i)/\Omega_{ii}^{1/2})),$$

where ζ is uniform on (0,1), and then drawing

$$(13) \quad V_{-i} = \mu_{-i} + \Omega_{-i,i} \Omega_{ii}^{-1} (V_i - \mu_i) + L \eta_{-i},$$

where L is a Choleski factor of $\Omega_{-i,-i} - \Omega_{-i,i} \Omega_{ii}^{-1} \Omega_{i,-i}$. Choosing the i that minimizes $E_V \mathbf{1}(V \in A)$ then provides a practical bound.

A continuous-in-parameters simulator of $1/P$ can be constructed, at the cost of some bias, using a variant of Lemma 3 suggested by Ruud:

Lemma 4. *Make the assumptions of Lemma 3, except take*

$$(14) \quad R = 1 + \sum_{i=1}^{r-1} \prod_{j=1}^i Q_j,$$

with the Q_j smooth independent unbiased simulators of $\mathbf{E}_{V|A} \mathbf{1}(V \notin B)$. Then

$$(15) \quad 1/P - (ER)/\mathbf{E}_V \mathbf{1}(V \in A) = (1 - \mathbf{E}_{V|A} \mathbf{1}(V \in B))^r / P.$$

Proof: From the proof of Lemma 3,

$$ER - 1/\mathbf{E}_{V|A} \mathbf{1}(V \in B) = \sum_{i=r}^{\infty} \prod_{j=1}^i \mathbf{E} Q_j = [\mathbf{E}_{V|A} \mathbf{1}(V \notin B)]^r / \mathbf{E}_{V|A} \mathbf{1}(V \in B). \quad \square$$

Suppose one simulates $1/P$ by a ratio $N/\tau \mathbf{E}_V \mathbf{1}(V \in A)$, where N is the random number of sequential draws necessary to obtain τ occurrences of the event $V \in B$ given $V \in A$, where the notation is as in Lemma 2. The following result due to McFadden, applied with $U_i = 1/\mathbf{E}_V \mathbf{1}(V \in A)$ and $W_i = \mathbf{1}(V \in B)$, with V drawn from the density $f(v|A)$, bounds the bias in this simulator:

Lemma 5. *Suppose (U_i, W_i) are independently identically distributed with $U_i \geq 0$, $1 \geq W_i$, $\mathbf{E} W_i = \beta > 0$, and $\mathbf{E} U_i = \alpha > 0$. Let $\tau > 1$ denote a positive constant. Suppose N is an integer-valued random variable defined as the first n such that $\sum_{i=1}^n W_i \geq \tau$. Then*

$$(16) \quad \left| \mathbf{E} \sum_{i=1}^N U_i / \sum_{i=1}^N W_i - \alpha/\beta \right| \leq \tau^{-1} \alpha/\beta.$$

Proof: A lemma of Stein's (Siegmund (1985), Proposition (2.19)) establishes $P(N > n) < C\rho^n$, for constants $C > 0$ and $0 < \rho < 1$, implying $\mathbf{E} N < +\infty$. Then, Wald's identity (Siegmund (1985), Proposition (2.20)) establishes $\mathbf{E} \sum_{i=1}^N U_i = \alpha \mathbf{E} N$ and $\mathbf{E} \sum_{i=1}^N W_i = \beta \mathbf{E} N$. Then, $\tau \leq \beta \mathbf{E} N < \tau + 1$, implying

$$\mathbf{E} \sum_{i=1}^n U_i / \sum_{i=1}^n W_i \leq \alpha \mathbf{E} N \tau < (\alpha/\beta)(1 + 1/\tau)$$

and

$$\mathbf{E} \sum_{i=1}^n U_i / \sum_{i=1}^n W_i \geq \alpha \mathbf{E} N / (m + 1) \geq (\alpha/\beta)(1 - 1/(\tau + 1)) > (\alpha/\beta)(1 - 1/\tau). \quad \square$$

This result can also be applied when correlated draws are used in the numerator and denominator of expressions like $(\nabla_{\mu} P)/P$.

2.6 Importance Sampling

Importance sampling is a general method for reducing the sampling variance of integrals computed by Monte Carlo integration. The CFS involves sampling V from the $n(\nu - \mu, \Omega)$ p.d.f. and evaluating the indicator $\mathbf{1}(V \in \mathbf{B})$. A simple generalization of this procedure rewrites the rectangle probability $P(\mathbf{B}; \mu, \Omega)$ in terms of another sampling distribution:

$$P(\mathbf{B}; \mu, \Omega) = \int \mathbf{1}(\nu \in \mathbf{B}) n(\nu - \mu, \Omega) d\nu = \int \left[\mathbf{1}(\nu' \in \mathbf{B}) \frac{n(\nu' - \mu, \Omega)}{f(\nu'; \mu, \Omega)} \right] f(\nu'; \mu, \Omega) d\nu'.$$

By sampling V' from the importance p.d.f. f and evaluating the weighted indicator function $\mathbf{1}(V' \in \mathbf{B}) n(V' - \mu, \Omega) / f(V'; \mu, \Omega)$, one obtains an alternative unbiased simulation of P . The first advantage that importance sampling offers is the ability to substitute sampling from f for sampling from n . It may be possible to sample f more quickly, or, in a more general setting, sampling from n may be infeasible. In addition, if f also has an analytical integral over a truncated sampling region \mathbf{C} such that $\mathbf{B} \subseteq \mathbf{C}$, then this analytical integral can be exploited as an approximation to P :

$$P(\mathbf{B}; \mu, \Omega) = \Pr\{V' \in \mathbf{C}\} \int_{\mathbf{C}} \left[\mathbf{1}(\nu' \in \mathbf{B}) \frac{n(\nu' - \mu, \Omega)}{f(\nu'; \mu, \Omega)} \right] \frac{f(\nu'; \mu, \Omega)}{\Pr\{V' \in \mathbf{C}\}} d\nu'.$$

By drawing from the truncated p.d.f. $f / \Pr\{\mathbf{C}\}$, fewer simulations are “wasted” on outcomes of zero and, in effect, $\Pr\{V' \in \mathbf{C}\} \frac{n(\nu' - \mu, \Omega)}{f(\nu'; \mu, \Omega)}$ approximates $P(\mathbf{B}; \mu, \Omega)$. When f is a good approximation to n , so that the ratio of densities n/f is relatively constant, the sampling variance of the importance-sampling simulator is small. The sampling variance of the CFS is $P(1 - P)$ and the sampling variance of the importance sampler is

$$\text{Var}(P_{\mathbf{C}} \cdot \mathbf{1}\{V' \in \mathbf{B}\} w(V')) = P_{\mathbf{C}}^2 \cdot P_{\mathbf{B}} \cdot \left[\text{V}(w(V') \mid V' \in \mathbf{B}) + (1 - P_{\mathbf{B}}) \cdot \text{E}(w(V') \mid V' \in \mathbf{B})^2 \right],$$

where $P_{\mathbf{C}} \equiv \Pr\{V' \in \mathbf{C}\}$ and $P_{\mathbf{B}} \equiv \Pr\{V' \in \mathbf{B}\}$. In the extreme case $f = n$ and $\mathbf{C} = \mathbf{B}$, the latter is zero; clearly good approximations to n afford improvements over the CFS.

2.7 Antithetic Variates

The accuracy of simulators may be improved, for a given number of Monte Carlo draws, by use of antithetic variates. The principle of antithetic variates, which is to introduce negative correlation between successive Monte Carlo draws to reduce the variance of simulation sample averages, generalizes in the multivariate case to selection of a regular grid of points whose location is random. This technique can be employed in simulators of H and $H_{\mathbf{C}}$ when the method has sufficient symmetry.

In Monte Carlo simulation requiring draws of m -dimensional standard normal vectors, for example the crude frequency simulator, an approach due to Deák (1980a,b) has good antithetic properties: Draw m independent standard normal vectors, and save their lengths ρ_1, \dots, ρ_m . Apply a Gram-Schmidt orthonormalization to these vectors to create a random basis for m -space. Let v_1, \dots, v_m denote the unit vectors in this basis. Each pair of unit vectors v_i and v_j with $i \neq j$ define a great circle on the unit hypersphere. Generate $2m$ equally spaced points on each great circle,

$$v_{i;k} = v_i \cos(\pi k/m) + v_j \sin(\pi k/m), \quad k = 0, \dots, 2m - 1.$$

This gives a random antithetic grid on the unit hypersphere that is particularly useful for simulations based on spherical transformations. Scale each unit vector v_{ijk} by the lengths ρ_1, \dots, ρ_m to give an array of normal vectors in m -space. A further antithetic refinement uses the fact that the squared lengths of standard normal vectors in m -space have a chi-squared distribution with m degrees of freedom. Taking $\rho_j^2 = \text{invchi}((\zeta + j)q, m)$, $j = 0, \dots, q - 1$, where ζ is a uniform $[0,1]$ random number and $\text{invchi}(\cdot, m)$ is the inverse of the chi-squared distribution, gives q antithetic lengths that are images of a random grid from the uniform distribution.

2.8 Control Variates

Control variates are random variables with analytic expectations that are positively correlated with the random variable whose expectation is to be simulated. Then, simulation variance can be reduced by simulating only the difference between the expectations of the variate of interest and the control variate. A possible control variate for $1(V \in B)h(V)$ is $1(V \in A)h(V)$, where A is the event $a_i < V_i < b_i$, with i a specific component of V . Then, for any α ,

$$(17) \quad H = \alpha E_V 1(V \in A)h(V) + E_V [1(V \in B) - \alpha 1(V \in A)]h(V).$$

The first expectation on the right-hand-side of (17) has a closed form, given below. The second expectation requires simulation. This control variate is not guaranteed to reduce simulation noise, but generally simulations based on (17) will be more accurate than simulations based on (6) if α is a moderately good preliminary estimate of $[E_V 1(V \in A)h(V)]^{-1} E_V 1(V \in B)h(V)$.

To obtain the analytic expression for $E_V 1(V \in A)h(V)$, first define the univariate standard normal partial moment

$$(18) \quad \Phi(t, k, \kappa) = \int_{-\infty}^t (v - \kappa)^k \phi(v) dv.$$

Integration by parts yields the recursion formulas

$$(19) \quad \begin{aligned} \Phi(t, 0, \kappa) &= \Phi(t), \\ \Phi(t, 1, \kappa) &= -\kappa \Phi(t) - \phi(t) \\ \Phi(t, k, \kappa) &= (k-1)\Phi(t, k-2, \kappa) - \kappa \Phi(t, k-1, \kappa) - (t-\kappa)^{k-1} \phi(t), \end{aligned}$$

for $k > 1$.

Now consider a multivariate normal density $n(v - \mu, \Omega)$ of dimension m , written as the product of a univariate marginal density $n(y - \mu_y, \Omega_{yy})$ and a dimension $m - 1$ conditional density $n(z - \mu_z - \Omega_{zz}^{-1} \Omega_{zy}(v_y - \mu_y), \Omega_{zz} - \Omega_{zy} \Omega_{yy}^{-1} \Omega_{yz})$. Consider the array of partial moments

$$(20) \quad \Psi(t, k, \mu, \Omega) = \int_{-\infty}^{+\infty} 1(y < t) y^k \begin{bmatrix} 1 & z - \mu_z & y - \mu_y \\ (z - \mu_z)' & (z - \mu_z)'(z - \mu_z) - \Omega_{zz} & (z - \mu_z)'(y - \mu_y) - \Omega_{zy} \\ y - \mu_y & (y - \mu_y)(z - \mu_z) - \Omega_{yz} & (y - \mu_y)^2 - \Omega_{yy} \end{bmatrix}$$

$$\cdot n(z - \mu_z - (y - \mu_y)\Omega_{yy}^{-1}\Omega_{yz}, \Omega_{zz} - \Omega_{zy}\Omega_{yy}^{-1}\Omega_{yz})n(y - \mu_y, \Omega_{yy})dzdy \equiv \begin{bmatrix} \Psi_1 & \Psi_z & \Psi_y \\ \Psi'_z & \Psi_{zz} & \Psi_y \\ \Psi_y & \Psi_{yz} & \Psi_{yy} \end{bmatrix}.$$

Then, letting $\Phi_j = \Omega_{yy}^{1/2}\Phi((t - \mu_y)/\sqrt{\Omega_{yy}}, j, -\mu_y/\sqrt{\Omega_{yy}})$,

$$(21) \quad \Psi_1 = \Phi_k, \Psi_y = \Phi_{k+1} - \mu_y\Phi_k, \Psi_{yy} = \Phi_{k+2} - 2\mu_y\Phi_{k+1} + (\mu_y^2 - \Omega_{yy})\Phi_k,$$

$$\Psi_z = \Psi_y\Omega_{yy}^{-1}\Omega_{yz}, \Psi_{yz} = \Psi_{yy}\Omega_{yy}^{-1}\Omega_{yz}, \Psi_{zz} = \Omega_{zy}\Omega_{yy}^{-1}\Psi_{yy}\Omega_{yy}^{-1}\Omega_{yz}.$$

Then, $\Psi(b, 0, \mu, \Omega) - \Psi(a, 0, \mu, \Omega)$, with y corresponding to component i of V , gives $E_V \mathbf{1}(V \in A)h(V)$.

2.9 Crude Frequency Simulator (CFS)

Recall equation (9), which states that the random vector V can be written as $V = \mu + \Gamma\eta$, where η is an independent standard normal vector of dimension m and Γ is a lower triangular Choleski factor of Ω , so $\Omega = \Gamma\Gamma'$. The *crude frequency simulator* (CFS) of $P(B; \mu, \Omega)$ and its derivatives is to make repeated Monte Carlo draws for η , use (9) to calculate V for each parameter vector, and then form an empirical analogue of the expectation in (6).⁸ A crude frequency simulator for H_C can be formed by rejecting draws of V that do not satisfy the conditioning event $V \in B$, and then forming an empirical analogue of the conditional expectation in (7) using the accepted draws. The CFS are quick to compute and ideal for parallel processing. However, they are *not* continuous in parameters, exhibiting jumps at parameter values yielding draws of V on the boundary of B .⁹ These discontinuities, however, can slow iterative parameter search.¹⁰ The accuracy of the CFS can be improved by use of antithetic and control variates introduced later.

2.10 Normal Importance Sampling (NIS)

Suppose the integrand $h(v)n(v - \mu, \Omega)$ in H can, with multiplication and division by a factor if necessary, be written as the product of a density $\gamma(v)$, whose support coincides with or contains B , from which it is easy to sample, and the remainder. Then, H can be written

$$(22) \quad H = \int \{\mathbf{1}(v \in B)h(v)n(v - \mu, \Omega)/\gamma(v)\}\gamma(v)dv \equiv E_\gamma \mathbf{1}(v \in B)h(v)n(v - \mu, \Omega)/\gamma(v).$$

An empirical expectation using draws from $\gamma(v)$ gives an unbiased simulator that is smooth in parameters, provided $h(v)n(v - \mu, \Omega)/\gamma(v)$ is dominated by a function whose expectation with respect to the density γ exists. We term this *Normal Importance Sampling* (NIS). Importance

⁸Frequency simulation of probabilities is a traditional method in numerical analysis; Lerman and Manski (1981) introduced this approach to estimation of multinomial probit models.

⁹These discontinuities do not prevent use of these simulators for statistical inference. If η is not redrawn when parameters change so that "chatter" is avoided, then these simulators are piecewise constant in parameters, and the manifolds on which discontinuities occur are linear. These properties imply a stochastic equicontinuity property that is sufficient to make the simulators well-behaved in statistical inference; see McFadden (1989).

¹⁰See Quandt (1984) for a discussion of iterative parameter search algorithms.

sampling works well when $h(v)n(v - \mu, \Omega)/\gamma(v)$ is fairly flat, and $\gamma(v)$ concentrates probability in the same region as $n(v - \mu, \Omega)$.

For fast computation, choose γ so that the components are independent, or are obtained as simple transformations of independent variates. A possible choice of γ when the rectangle $[a, b]$ is finite is the uniform distribution on this rectangle. Independent exponential densities,

$$(23) \quad \gamma(v) = \prod_{i=1}^m \exp(-(v_i - b_i)/c_i)/c_i,$$

where c_i are parameters that can be set as part of the simulation, are feasible; draws from this density are easily computed using $v_i = b_i + c_i \cdot \log \zeta_i$, where ζ_i is a uniform $[0,1]$ variate. This particular choice of $\gamma(v)$ defines the NISE simulator.¹¹ An alternative that is more likely to concentrate probability for γ in the same region as the multivariate normal is the product of truncated normals,

$$(24) \quad \gamma(v) = \prod_{i=1}^m \phi((v_i - \alpha_i)/c_i) / [\Phi((b_i - \alpha_i)/c_i) - \Phi((a_i - \alpha_i)/c_i)], \quad v \in B,$$

with $\alpha_i = \mu_i$ and $c_i = \sqrt{\Omega_{ii}}$. One can sample from this distribution using

$$(25) \quad v_i = \alpha_i + c_i \Phi^{-1}(\zeta_i \Phi((b_i - \alpha_i)/c_i) + (1 - \zeta_i) \Phi((a_i - \alpha_i)/c_i)),$$

with ζ_i a uniform $[0,1]$ variate. Using $\gamma(v)$ given in (24) defines the NIST simulator.¹² Reduction in simulation variance is in principle achievable using antithetic variates, with the ζ_i in dimension i drawn in repeated trials from a random grid on the unit interval. A variation on (24) for finite rectangles $[a, b]$ is

$$(26) \quad \alpha_i = \mu_i + \Omega_{i,-i} \Omega_{-i,-i}^{-1} (b_{-i} + a_{-i})/2$$

$$(27) \quad c_i^2 = \Omega_{ii} - \Omega_{i,-i} \Omega_{-i,-i}^{-1} \Omega_{-i,i}.$$

This takes some account of the shift in location of probability mass in the multivariate normal density induced by correlation and restriction to B .

2.11 Kernel-Smoothed Frequency Simulator (KFS)

The Kernel Smoothed Frequency Simulator (KFS), suggested by McFadden (1989), replaced the indicator function $1(v \in B)$ in the crude frequency simulator with the function

$$(28) \quad \mathcal{K}_{B,\omega}(v) \equiv \mathcal{K}((v - b)/\omega) - \mathcal{K}((v - a)/\omega),$$

where $\mathcal{K}(w)$ is a smooth kernel function from \mathbf{R}^m onto $[0,1]$ with $\mathcal{K}(-\infty) = 1$ and $\mathcal{K}(+\infty) = 0$, and ω is a window width parameter. The function $\mathcal{K}_{B,\omega}(v)$ approaches $1(v \in B)$ as $\omega \rightarrow 0$. Then,

¹¹ "E" for exponential.

¹² "T" for truncated normals.

the simulator is an average of $\mathcal{K}_{\mathbf{B},\omega}(v)h(v)$, with $v = \mu + \Gamma\eta$, over τ Monte Carlo draws of an independent standard normal vector η . This simulator is smooth in parameters, a useful feature when the simulator is used within an iterative optimization. The simulator is a biased estimate of H for positive ω , but in statistical applications one can shrink ω as sample size increases:

Lemma 6. Define $\mathbf{C}_\lambda = \{w \in \mathbf{R}^m \mid |w_i| \geq \lambda \text{ for all } i\}$. Suppose there exist λ_0 , K , and $q > 0$ such that for $\lambda \geq \lambda_0$,

$$(29) \quad \sup_{w \in \mathbf{C}_\lambda} |\mathbf{1}(w < 0) - \mathcal{K}(w)| \leq K\lambda^{-q}.$$

Then there exists a constant $K' > 0$ such that for small ω ,

$$(30) \quad \left| \int [\mathcal{K}_{\mathbf{B},\omega}(v) - \mathbf{1}(v \in \mathbf{B})] h(v) n(v - \mu, \Omega) dv \right| < K' \omega^{q/(1+q)}.$$

Proof: The conditional expectation of $h(V)$ with respect to the distribution of V_- ; given $V_i = v_i$ is bounded by a function $K_1 + K_2(v_i - \mu_i)^2$ for some constants K_1, K_2 . Then the integral of $[\mathcal{K}((v-b)/\omega) - \mathbf{1}(v < b)]h(v)n(v - \mu, \Omega)$ over the region $(v-b)/\omega \in \mathbf{C}_\lambda$ is bounded by $K\lambda^{-q}tr(\Omega)$, and over the region $(v-b)/\omega \notin \mathbf{C}_\lambda$ is bounded by

$$\begin{aligned} & \sum_{i=1}^m \int_{|v_i - b_i| < \lambda\omega} n(v_i - \mu_i, \Omega_{ii}) [K_1 + K_2(v_i - \mu_i)^2] dv_i \\ & \leq \sum_{i=1}^m 2\lambda\omega \Omega_{ii}^{-1} [K_1 + K_2((b_i - \mu_i)^2 + (\lambda\omega)^2)]. \end{aligned}$$

Take $\lambda = \omega^{-1/(1+q)}$ and combine the regions to conclude that the integral is bounded by $\omega^{q/(1+q)}K'/2$ for some constant K' . Repeat the argument, replacing b by a , to obtain the bound (30). \square

Ruud has suggested starting iterative parameter search with a large ω , and shrinking it as the search approaches convergence. This avoids many of the problems of handling discontinuities in search.

A variety of kernel functions \mathcal{K} can be employed such as a logistic function $\mathcal{K}(v) = 1/(1 + e^{v_1} + \dots + e^{v_m})$, a normal kernel $\mathcal{K}(v) = \prod_{i=1}^m \Phi(-v_i)$, or a polynomial kernel $\mathcal{K}(v) = \prod_{i=1}^m G(v_i)$, where

$$(31) \quad G(y) = \begin{cases} 1 & \text{for } y \leq -1 \\ (1 - y(2 + y))/2 & \text{for } -1 < y \leq 0 \\ (1 - y(2 - y))/2 & \text{for } 0 < y < 1 \\ 0 & \text{for } y \geq 1. \end{cases}$$

The polynomial kernel is particularly quick to compute, and will generally be the method of choice. As in the case of the crude frequency simulator, antithetic draws can be used for η in (6). It is also possible in the case of the polynomial kernel to use a smoothed version $[G((v_i - b)/\omega) -$

$G((v_i - a)/\omega)h(v)$ of the control variable $1(a_i < v_i < b_i)h(v)$. Let y denote v_i , and z denote v_{-i} . Then, $\Upsilon(\omega, b, \mu, \Omega) - \Upsilon(\omega, a, \mu, \Omega)$ is the expectation of the smoothed control variate, where

$$(32) \quad \Upsilon(\omega, t, \mu, \Omega) = \int_{-\infty}^{+\infty} G((y-t)/\omega) \begin{bmatrix} 1 & z - \mu_z & y - \mu_y \\ (z - \mu_z)' & (z - \mu_z)'(z - \mu_z) - \Omega_{zz} & (z - \mu_z)'(y - \mu_y) - \Omega_{zy} \\ y - \mu_y & (y - \mu_y)(z - \mu_z) - \Omega_{yz} & (y - \mu_y)^2 - \Omega_{yy} \end{bmatrix} \cdot n(z - \mu_z - (y - \mu_y)\Omega_{yy}^{-1}\Omega_{yz}, \Omega_{zz} - \Omega_{zy}\Omega_{yy}^{-1}\Omega_{yz})n(y - \mu_y, \Omega_{yy})dzdy.$$

Breaking up the domain of integration, one has, from (20),

$$\begin{aligned} \Upsilon(\omega, t, \mu, \Omega) &= [(t - \omega)^2\Psi(t - \omega, 0, \mu, \Omega) - 2t^2\Psi(t, 0, \mu, \Omega) + (t + \omega)^2\Psi(t + \omega, 0, \mu, \Omega)]/2\omega^2 \\ &\quad - [(t - \omega)\Psi(t - \omega, 1, \mu, \Omega) - 2t\Psi(t, 1, \mu, \Omega) + (t + \omega)\Psi(t + \omega, 1, \mu, \Omega)]/\omega^2 \\ &\quad + [\Psi(t - \omega, 2, \mu, \Omega) - 2\Psi(t, 2, \mu, \Omega) + \Psi(t + \omega, 2, \mu, \Omega)]/2\omega^2. \end{aligned}$$

2.12 Stern Decomposition Simulator (SDS)

The Stern Decomposition, suggested by Stern (1992), writes $V \sim \mathcal{N}(\cdot; \mu, \Omega)$ as a sum $V = Y + W$, with $Y \sim \mathcal{N}(\cdot; \mu, \lambda^2 I)$ with $W \sim \mathcal{N}(\cdot; 0, \Omega - \lambda^2 I)$. That is, V equals the sum of a “small” independently distributed normal vector and a second normal vector that carries the information on the covariance matrix of V . Then, by the law of iterated expectations,

$$(33) \quad H = \int_w \left\{ \int_{y=a}^b h(y) \left[\prod_{i=1}^m \phi((y_i - w_i - \mu_i)/\lambda)/\lambda \right] dy \right\} n(w, \Omega - \lambda^2 I)dw.$$

The term in braces can be integrated analytically; then the SDS averages this interior integral over r Monte Carlo draws $w = [\Omega - \lambda^2 I]^{1/2}\eta$, where η is a standard normal vector and $[\Omega - \lambda^2 I]^{1/2}$ is a Choleski factor of $\Omega - \lambda^2 I$. The array H can also be written, by multiplying and dividing by the density $n(v, \Omega - \lambda^2 I)$, as

$$(34) \quad H = C \cdot \int_v \left\{ 1(v \in B) \exp\{-\lambda^2 v'(\Omega - \lambda^2 I)^{-1}\Omega^{-1}v/2 + \mu'\Omega^{-1}v\} \right\} n(v, \Omega - \lambda^2 I)dv,$$

where $C = \exp(-\mu'\Omega^{-1}\mu/2)[\det(\Omega - \lambda^2 I)/\det(\Omega)]^{1/2}$. There are two possible interpretations of this simulator. First, (33) can be viewed as a kernel-smoothed variant of the integrand in (34). Second, (34) can be interpreted as a form of importance sampling, with $n(\cdot, \Omega - \lambda^2 I)$ the convenient density from which Monte Carlo draws are made.

To obtain the analytic form for the term in braces in (33), define

$$(35) \quad q_{ik} = \lambda^k \left[\Phi\left(\frac{b_i - \mu_i - w_i}{\lambda}, k, -\frac{2w_i}{\lambda}\right) - \Phi\left(\frac{a_i - \mu_i - w_i}{\lambda}, k, -\frac{2w_i}{\lambda}\right) \right],$$

where $\Phi(t, k, \kappa)$ is the partial moment function (19). Define

$$(36) \quad Q = \prod_{i=1}^m q_{i0},$$

$$(37) \quad s = (q_{11}/q_{10}, \dots, q_{m1}/q_{m0})',$$

and

$$(38) \quad D = \text{diag}\{q_{i2}/q_{i0}\}.$$

Then, (33) becomes

$$(39) \quad H = \int_w Q \cdot \begin{bmatrix} 1 & s'\Omega^{-1} \\ \Omega^{-1}s & \frac{1}{2}\Omega^{-1}[D + s \cdot s' - \Omega]\Omega^{-1} \end{bmatrix} n(w, \Omega - \lambda^2 I) dw.$$

The SDS provides an unbiased smooth simulator. Again, accuracy can be enhanced using antithetics when drawing η to construct the simulated values $w = (\Omega - \lambda^2 I)^{1/2} \eta$. This simulator is fast to compute, but it can be computationally burdensome to determine λ such that $\Omega - \lambda^2 I$ is positive definite, and accuracy fails when m is large and the eigenvalues of Ω are uneven. The extension of the Stern simulator to derivatives is due to McFadden.

2.13 Geweke-Hajivassiliou-Keane Simulator (GHK)

This simulator is based on sampling from recursive truncated normals after a Choleski transformation. The approach was suggested by Geweke (1989), and has been developed by Hajivassiliou, who proposed the weighting used here. Keane (1994) independently developed a weighting scheme of essentially the same form for a problem of estimating transition probabilities.

Consider the triangularizing transformation $V = \mu + \Gamma\eta$, where Γ is the Choleski factor of Ω . The indicator $1(v \in B)$ is then transformed to $1(\mu + \Gamma\eta \in B)$, which can be written recursively as the product of indicators of the events

$$(40) \quad \mathbf{B}_j(\eta_{<j}) = \{\eta_j | (a_j - \mu_j - \Gamma_{j,<j}\eta_{<j})/\Gamma_{jj} < \eta_j < (b_j - \mu_j - \Gamma_{j,<j}\eta_{<j})/\Gamma_{jj}\}$$

for $j = 1, \dots, m$. Define $\phi(\eta_j | \mathbf{B}_j(\eta_{<j})) = \phi(\eta_j)1(\eta_j \in \mathbf{B}_j(\eta_{<j}))/\Phi(\mathbf{B}_j(\eta_{<j}))$, the conditional distribution of η_j given the event $\mathbf{B}_j(\eta_{<j})$. Define a weight

$$(41) \quad \omega(\eta) = \prod_{j=1}^m \phi(\eta_j | \mathbf{B}_j(\eta_{<j})).$$

Then

$$(42) \quad H = \int h(\mu + \Gamma\eta)\omega(\eta) \prod_{j=1}^m \phi(\eta_j | \mathbf{B}_j(\eta_{<j})) d\eta.$$

The following result gives the GHK simulator:

Lemma 7. *An unbiased simulator of H is an average of $h(\mu + \Gamma\eta)\omega(\eta)$, where $\omega(\eta)$ is the weighting function (41), over draws constructed recursively from the one-dimensional conditional densities $\phi(\eta_j | \mathbf{B}_j(\eta_{<j}))$ by taking*

$$(43) \quad \eta_j = \Phi^{-1}(\zeta_j \Phi((a_j - \mu_j - \Gamma_{j,<j}\eta_{<j})/\Gamma_{jj}) + (1 - \zeta_j)\Phi((b_j - \mu_j - \Gamma_{j,<j}\eta_{<j})/\Gamma_{jj})),$$

where the ζ_j are draws from the uniform $[0,1]$ density. When \mathbf{B} is a finite rectangle, there is a constant $c > 0$ such that for each ϵ , $0 < \epsilon < 1$,

$$(44) \quad \text{Prob} \left(\frac{\sum_{i=1}^r h(\mu + \Gamma \eta_i) \omega(\eta_i)}{\sum_{i=1}^r \omega(\eta_i)} - H_C > \epsilon \right) < \exp(-rc\epsilon^2).$$

Proof: It is immediate from (42) that the GHK simulator of H is unbiased. Now consider the simulator of H_C in (44). Write $A = \{\eta | \mu + \Gamma \eta \in \mathbf{B}\}$ and let $\tilde{\eta}$ denote a random vector in A drawn recursively using (43). Then,

$$\begin{aligned} \mathbf{E}_{V|V \in \mathbf{B}} h(V) &= \mathbf{E}_{\eta|\eta \in A} h(\mu + \Gamma \eta) = \frac{\int_{\eta \in A} h(\mu + \Gamma \eta) \prod_{k=1}^m \phi(\eta_k) d\eta_k}{\prod_{k=1}^m \phi(\eta_k) d\eta_k} \\ &= \frac{\int_{\eta \in A} h(\mu + \Gamma \eta) \omega(\eta) \prod_{k=1}^m [\phi(\eta_k) / \Phi(\mathbf{B}_k(\eta_{<k}))] d\eta_k}{\int_{\eta \in A} \omega(\eta) \prod_{k=1}^m [\phi(\eta_k) / \Phi(\mathbf{B}_k(\eta_{<k}))] d\eta_k} \\ &= [\mathbf{E}_{\tilde{\eta}} h(\mu + \Gamma \tilde{\eta})] / \mathbf{E}_{\tilde{\eta}} \omega(\tilde{\eta}). \end{aligned}$$

To obtain the bound (44) when \mathbf{B} is finite, note that in this case there is a bound c_1 on every component of the array $h(\mu + \Gamma \eta) \omega(\eta)$. Then, by Hoeffding's inequality,

$$(45) \quad \text{Prob} \left(\frac{1}{r} \sum_{i=1}^r h(\mu + \Gamma \eta_i) \omega(\eta_i) - H > \epsilon_1 \right) < \exp[-r\epsilon_1^2 / 2c_1^2].$$

Choose $\epsilon_1 = \epsilon P^2 / (2 + c_1)$. Suppose the event in (45) does not occur. Then,

$$\begin{aligned} \frac{\sum_{i=1}^r h(\mu + \Gamma \eta_i) \omega(\eta_i)}{\sum_{i=1}^r \omega(\eta_i)} - H_C &\leq \frac{\frac{1}{r} \sum_{i=1}^r h(\mu + \Gamma \eta_i) \omega(\eta_i) - H}{\frac{1}{r} \sum_{i=1}^r \omega(\eta_i)} + |H_C| \cdot \frac{\frac{1}{r} \sum_{i=1}^r \omega(\eta_i) - P}{\frac{1}{r} \sum_{i=1}^r \omega(\eta_i)} \\ &\leq \epsilon_1 / (P - \epsilon_1) + c_1 \epsilon_1 / P(P - \epsilon_1) \leq \epsilon, \end{aligned}$$

so that the event in (44) does not occur. Therefore, letting $c = P^4 / 2c_1^2 (2 + c_1)^2$, the probability of the event in (44) is at most $\exp(-rc\epsilon^2)$. \square

The GHK simulator can be interpreted as importance sampling using the recursive truncated standard normal densities. It has proven fast, with low noise, even when probabilities are small. Börsch-Supan and Hajivassiliou (1993) use this simulator¹³ to define the method of smoothly simulated maximum likelihood and show that this method exhibits excellent properties for limited dependent variable models that are otherwise difficult to estimate. In principle, it is possible to use antithetic variates in (43), starting from a random grid of size q drawn from the uniform distribution in each step of the recursion. In practice, this produces q^m points, which is impractically large in many problems.¹⁴

¹³which they term the "Smooth, Recursive Conditioning" simulator

¹⁴Some of the desirable features of antithetic variates can be obtained by sampling without replacement from a large random grid.

2.14 Parabolic Cylinder Function Simulator (PCF)

Consider a spherical transformation $V = b + \rho \cdot v$ about the upper limit b of B , where v is in the unit sphere and ρ is a non-negative scalar. The Jacobean of the transformation from V to $(v_1, \dots, v_{m-1}, \rho)$ is ρ^{m-1}/v_m , where $v_m = [1 - v_1^2 - \dots - v_{m-1}^2]^{1/2}$. Then, the multivariate normal density $n(V - \mu, \Omega)$ expressed in terms of the transformed variables is

$$f(\rho|v)q(v|\mu - b, \Omega) = (2\pi)^{-m/2}|\Omega|^{-1/2} \exp[-(\rho v - (\mu - b))'\Omega^{-1}(\rho v - (\mu - b))/2]\rho^{m-1}/v_m.$$

In this expression, the conditional density of ρ given v is

$$(46) \quad f(\rho|v) = \rho^{m-1} \exp[-(\rho - (\mu - b))'\Omega^{-1}v/v'\Omega^{-1}v)^2(v'\Omega^{-1}v)/2]/K(v),$$

with

$$K(v) = \int_0^\infty \rho^{m-1} \exp[-(\rho - (\mu - b))'\Omega^{-1}v/v'\Omega^{-1}v)^2(v'\Omega^{-1}v)/2]d\rho,$$

and the marginal distribution of v is

$$q(v|\mu - b, \Omega) = K(v)(2\pi)^{-m/2}|\Omega|^{-1/2} \exp[-(\mu - b)'\Omega^{-1}(\mu - b)/2 + (v'\Omega^{-1}(\mu - b))^2/2v'\Omega^{-1}v]/v_m.$$

When $\mu - b = 0$ and $\Omega = I$, so that the distribution of v on the unit sphere is uniform, $q(v|0, I) = (2\pi)^{-m/2} \int_0^\infty \rho^{m-1} \exp[-\rho^2/2]d\rho/v_m = \pi^{-m/2}\Gamma(m/2)/2v_m$. Define an importance sampling weight

$$(47) \quad s(v) \equiv q(v|\mu - b, \Omega)/q(v|0, I)K(v) \\ = \exp[-(\mu - b)'\Omega^{-1}(\mu - b)/2 + (v'\Omega^{-1}(\mu - b))^2/2v'\Omega^{-1}v]/\Gamma(m/2)2^{m/2-1}|\Omega|^{1/2}.$$

Define

$$(48) \quad \bar{\rho}(v) = \min_j n(a_j - b_j)/v_j,$$

and note that it is positive for $v < 0$. Then,

$$(49) \quad H = \int_{v < 0} \left(\int_{\rho=0}^{\bar{\rho}(v)} h(b + \rho v) f(\rho|v) d\rho \right) K(v) s(v) q(v|0, I) dv.$$

Define

$$(50) \quad C(i, \alpha, \beta, \gamma) = \int_0^\gamma \rho^i \exp[-\alpha(\rho - \beta/\alpha)^2/2] d\rho.$$

This function satisfies

$$(51) \quad C(0, \alpha, \beta, \gamma) = (2\pi/\alpha)^{1/2} [\Phi((\alpha\gamma - \beta)/\sqrt{\alpha}) - \Phi(-\beta/\sqrt{\alpha})],$$

$$(52) \quad C(1, \alpha, \beta, \gamma) = (\exp[-\beta^2/2\alpha] - \exp[-\alpha(\gamma - \beta/\alpha)^2/2])/\alpha + C(0, \alpha, \beta, \gamma)\beta/\alpha,$$

$$(53) \quad C(i, \alpha, \beta, \gamma) = C(i-1, \alpha, \beta, \gamma)\beta/\alpha + C(i-2, \alpha, \beta, \gamma)(i-1)/\alpha \\ - \gamma^{i-1} \exp[-\alpha(\gamma - \beta/\alpha)^2/2]/\alpha,$$

with (53) obtained by integration by parts for $i > 1$. Then, C can be computed by recursion in i . We term C a parabolic cylinder function, as it is closely related to a standard class of mathematical functions with this name; see Abramowitz and Stegun (1964, p. 685) and Spanier and Oldham (1987). When $\beta = 0$, this function is proportional to a chi-square cumulative distribution function,

$$C(i, \alpha, 0, \gamma) = 2^{(i-1)/2} \cdot \Gamma((i+1)/2) \cdot \alpha^{-(i+1)/2} \cdot \chi_{i+1}^2(\alpha\gamma^2).$$

The function $K(v)$ in (46) satisfies $K(v) = C(m-1, v'\Omega^{-1}v, (\mu-b)'\Omega^{-1}v, \infty)$. Define

$$(54) \quad C_i(v) \equiv K(v) \int_{\rho=0}^{\bar{\rho}(v)} \rho^i f(\rho|v) d\rho = C(i+m-1, v'\Omega^{-1}v, (\mu-b)'\Omega^{-1}v, \bar{\rho}(v)).$$

Then $H = \begin{bmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{bmatrix}$ satisfies

$$(55) \quad H_{11} = E_v C_0(v) s(v),$$

$$H_{21} = H'_{12}, = \Omega^{-1} E_v [C_1(v)v + (b-\mu)C_0(v)] s(v),$$

$$H_{22} = \frac{1}{2} \Omega^{-1} E_v [C_2(v)vv' - C_1(v) \{v(\mu-b)' + (\mu-b)v'\} + C_0(v) \{(\mu-b)(\mu-b)' - \Omega\}] s(v)\Omega^{-1},$$

where the expectation is with respect to the uniform density on the unit sphere. The PCF simulator is obtained by averaging the array (55) over a sample of draws of v from the uniform distribution on the unit sphere intersecting the negative orthant, and can be interpreted as importance sampling with this comparison distribution. The draws can be made by first drawing a standard normal vector η , reversing the signs of positive components so that it is negative, and then defining $v = \eta/\sqrt{\eta'\eta}$. This has the effect of drawing from $q(v|0, I)$ conditioned on $v < 0$. To remove this conditioning, the simulation average must be divided by 2^m , the number of orthants. This simulator was suggested by McFadden (1989).

2.15 Deák Chi-Square Simulator (DCS)

A spherical transformation about the mean of the multivariate distribution is $V = \mu + \rho \cdot v$, where again ρ is a scalar and v is a point in the unit sphere. We use the notation of the previous section, with μ replacing b , so that $s(v) = 1/2^{m/2-1} \Gamma(m/2) |\Omega|^{1/2}$ and, analogously to $C_i(v)$,

$$(56) \quad D_i(v) \equiv K(v) \int \mathbf{1}(\mu + \rho v \in B) \rho^i f(\rho|v) d\rho.$$

This expression can be evaluated analytically using (46)–(53), yielding

$$D_i(v) = C(i+m-1, v'\Omega^{-1}v, 0, \rho'(v)) - C(i+m-1, v'\Omega^{-1}v, 0, \rho''(v)),$$

where $\rho'(v)$ and $\rho''(v)$ are the largest and smallest values of ρ satisfying the inequalities $a - \mu < \rho v < b - \mu$. Then, $0 \leq \rho''(v) \leq \rho'(v)$,

$$(57) \quad \rho'(v) = \min \left\{ \min_{v_j > 0} (b_j - \mu_j)/v_j, \min_{v_j < 0} (a_j - \mu_j)/v_j \right\}$$

$$\rho''(v) = \max \left\{ 0, \max_{v_j \geq 0} (a_j - \mu_j)/v_j, \max_{v_j < 0} (b_j - \mu_j)/v_j \right\},$$

where by convention $0/0 = 0$, and $0 = \rho''(v) = \rho'(v)$ whenever (57) fails to yield a non-degenerate interval in the positive half-line. Taking the expectation of $D_i(v)s(v)$ with respect to the uniform distribution on the unit sphere,

$$(58) \quad \mathbf{E}_V D_i(v)s(v) = K'(v'\Omega^{-1}v)^{-(m+i)/2} [\chi_{i+m}^2(v'\Omega^{-1}v\rho'(v)^2) - \chi_{i+m}^2(v'\Omega^{-1}v\rho''(v)^2)],$$

with $K' = 2^{i/2}\Gamma((m+i)/2)/\Gamma(m/2)|\Omega|^{1/2}$. Then

$$(59) \quad H = \begin{bmatrix} \mathbf{E}_v D_0(v)s(v) & \mathbf{E}_v D_1(v)v's(v)\Omega^{-1} \\ \Omega^{-1}\mathbf{E}_v D_1(v)vs(v) & \frac{1}{2}\Omega^{-1}\mathbf{E}_v [D_2(v)vv' - D_0(v)\Omega]s(v)\Omega^{-1} \end{bmatrix},$$

with (58) used to evaluate the terms. The DCS simulator is obtained by drawing an antithetic random grid of points v on the unit sphere, as described in paragraph *G*, and forming the empirical expectation corresponding to the array in (59). It is unbiased for H and smooth in parameters. This simulator for P is due to Deák (1980a,b); the extension to obtain the derivatives of P was suggested by Chunrong Ai.

2.16 Acceptance/Rejection Simulator (ARS)

An Acceptance/Rejection simulator calculates an empirical analogue of the expectation $H_C = \mathbf{E}_{V|V \in B} h(V)$ by constructing draws from the conditional distribution of V given B . The crudest form of ARS is to sample from the unconditional distribution of V using (9), reject points not in B until r accepted points are found, and form an average of $h(V)$ over the accepted points. More generally, the procedure in Lemma 1 permits improvement in the yield of the method by sampling from a comparison distribution that puts little or no weight outside B and has the property that the ratio of the comparison density to the target density is uniformly bounded above by a small number. Nevertheless, yields tend to be low, making computation heavy, for small P .

A point $V = \mu + \Gamma\eta$ may, for given η , move from the rejection to the acceptance region with small changes in the parameters, introducing discontinuities in the ARS. Using an approach suggested by Ruud, Hajivassiliou and McFadden (1990) show that the ARS nevertheless has a stochastic equicontinuity property that enables its use in statistical applications.

Possible comparison distributions for A/R sampling are independent exponential or truncated normal distributions in V -space, but greater yields can be obtained using the recursive truncated normal distribution (43) employed in the GHK simulator. The following lemmas give protocols for use of independent exponential or recursive truncated normal comparison distributions. Lemma 8 is due to McFadden (1989), Lemma 9 to Hajivassiliou and McFadden (1990).

Lemma 8. Draw $v_i = b_i + (\log \zeta_i)\lambda_i$, $i = 1, \dots, m$, where the λ_i are positive constants chosen for the simulation and the ζ_i are uniform $[0,1]$, and accept v if

$$\log \zeta_0 \leq -(v - \mu)' \Omega^{-1} (v - \mu) / 2 - \lambda'(v - \mu) - \lambda' \Omega \lambda / 2.$$

Then, the accepted V are distributed $n(v - \mu, \Omega)$ conditioned on $V \in \mathbf{B}$, and the average of (8) over the accepted points is an unbiased simulator of H_C .

Proof: Apply Lemma 1 to the density $\prod_{C=1}^m \lambda_i \exp[\lambda_i(v_i - b_i)]$ on $v \leq b$. □

A suitable choice for λ might be $\lambda_i = \sqrt{\Omega_{ii}}$.

Lemma 9. Draw η sequentially using

$$\eta_j = \Phi^{-1}(\zeta_j \Phi((a_j - \mu_j - \Gamma_{j,<j} \eta_{<j}) / \Gamma_{jj}) + (1 - \zeta_j) \Phi((b_j - \mu_j - \Gamma_{j,<j} \eta_{<j}) / \Gamma_{jj}))$$

and accept $v = \mu + \Gamma \eta$ if $\zeta_0 \leq \prod_{i=2}^m [\Phi(\mathbf{B}_i(\eta_{<i}) / \alpha_i)]$, where the ζ_i are uniform $[0,1]$ random variates, $\alpha_i = \Phi((b_i - a_i) / 2\Gamma_{ii}) - \Phi(-(b_i - a_i) / 2\Gamma_{ii}) \leq 1$, and

$$\mathbf{B}_i(\eta_{<i}) = \{\eta_i | a_i - \mu_i - \Gamma_{i,\leq i} \eta_{<i} / \Gamma_{ii} < \eta_i < (b_i - \mu_i - \Gamma_{i,<i} \eta_{<i}) / \Gamma_{ii}\}.$$

Then, the accepted V are distributed $n(v - \mu, \Omega)$ conditioned on $V \in \mathbf{B}$, and the average of (8) over the accepted points is an unbiased simulator of H_C .

Proof: The protocol samples sequentially from the recursive truncated normal density $g(\eta) = \prod_{i=1}^m \phi(\eta_i) / \Phi(\mathbf{B}_i(\eta_{<i}))$, with $\eta_i \in \mathbf{B}_i(\eta_{<i})$. For this density,

$$f(\eta) / g(\eta) = \prod_{i=1}^m \Phi(\mathbf{B}_i(\eta_{<i})),$$

implying

$$(60) \quad 1 \geq \alpha \geq \sup_{\eta \in \mathbf{B}} \prod_{C=1}^m \Phi(\mathbf{B}_i(\eta_{<i})) \geq \int \prod_{C=1}^m \Phi(\mathbf{B}_i(\eta_{<i})) \cdot g(\eta) d\eta \equiv P.$$

In particular,

$$(61) \quad \Phi(\mathbf{B}_1) = \Phi((b_1 - \mu_1) / \Gamma_{11}) - \Phi((a_1 - \mu_1) / \Gamma_{11}) \equiv \alpha_1,$$

and

$$(62) \quad \sup_{\eta_i \in \mathbf{B}_i(\eta_{<i})} \Phi(\mathbf{B}_i(\eta_{<i})) \leq \sup_z [\Phi((b_i - \mu_i - z) / \Gamma_{ii}) - \Phi((a_i - \mu_i - z) / \Gamma_{ii})],$$

$$\Phi((b_i - a_i) / 2\Gamma_{ii}) - \Phi(-(b_i - a_i) / 2\Gamma_{ii}) \equiv \alpha_i$$

and one can take $\alpha = \prod_{i=1}^m \alpha_i \leq 1$. The result then follows from Lemma 1. □

We term the acceptance/rejection variant that is defined by Lemma 8 the ARSE simulator, while we use ARSR to refer to the method of recursively sampling from truncated normal densities, as described in Lemma 9.¹⁵

¹⁵It may be possible to increase the acceptance level in Lemma 9 in special cases by considering the structure of \mathbf{B}

2.17 Gibbs Sampler Simulator (GSS)

The Gibbs Sampler Simulator for H_C is based on a Markov chain that utilizes computable univariate truncated normal densities to construct transitions, and has the desired truncated multivariate normal as its limiting distribution. The simulator was developed by Hajivassiliou, starting from stochastic relaxation methods studied by Geman and Geman (1984). The following result establishes the theoretical properties of the simulator:

Lemma 10. *Suppose \mathbf{B} is finite. Start from any $v^{(0)} \in \mathbf{B}$. Define a recursive procedure with steps $i = 1, \dots, m$ in rounds $j = 1, \dots, r$. Suppose at step i in round j , $v^{(j-1)}$ and $v_{<i}^{(j)}$ have been determined. Define*

$$(63) \quad v_i^{(j)} = \kappa_{ij} + \sigma_i \Phi^{-1}(\zeta_{ij} \Phi((b_i - \kappa_{ij})/\sigma_i) + (1 - \zeta_{ij}) \Phi((a_i - \kappa_{ij})/\sigma_i)),$$

where the ζ_{ij} are independent uniform $[0,1]$ variates,

$$(64) \quad \kappa_{ij} = \mu_i + \Omega_{i,-i} \Omega_{-i,-i}^{-1} \begin{bmatrix} v_{<i}^{(j)} - \mu_{<i}^{(j)} \\ v_{>i}^{(j-1)} - \mu_{>i}^{(j-1)} \end{bmatrix},$$

and

$$(65) \quad \sigma_i = [\Omega_{ii} - \Omega_{i,-i} \Omega_{-i,-i}^{-1} \Omega_{-i,i}]^{1/2}.$$

Note that $v \in \mathbf{B}$. Let $\mathbf{E}_{(r)}$ denote expectation with respect to the distribution of the vector V obtained after r rounds of (63). Then, there exists $\rho \in (0, 1)$ and $K > 0$ such that

$$(66) \quad |\mathbf{E}_{(r)} h(V) - H_C| < K \rho^r.$$

Proof: See Hajivassiliou and McFadden (1990).

2.18 Sequentially Unbiased Simulator (SUS)

A family of unbiased simulators of $H_C = H/P$ can be formed from independent unbiased simulators of H and of $1/P$. Lemma 3 describes an unbiased simulator for $1/P$: Let \mathbf{A} denote the event $a_i < V_i < b_i$, where i is a specified component of V . For $r \geq 0$ initial steps, independent unbiased simulators Q_j of $1 - Pr(\mathbf{B})/Pr(\mathbf{A})$ are constructed, using the analytic expression for $Pr(\mathbf{A})$ and any of the smooth unbiased simulators of $Pr(\mathbf{B})$ provided by NIS, SDS, GHK, PCF, or DCS. After the initial r steps, V_i are drawn, conditional on \mathbf{A} , as detailed in (12) and (13). This sampling proceeds sequentially, setting $Q_j = 1$, until the event $V_i \in \mathbf{B}$ is observed. Then (11) provides an unbiased simulator of $1/P$. The purpose of the smooth simulators in the first r steps is to reduce the variance in the simulation of $1/P$. For numerical purposes, this method coincides with the following simulator when an extremely large censoring point r , the predetermined limit for the number of steps, is used.

in more detail. For example, when the below-diagonal terms of Γ are all non-negative, the supremum in (62) can be defined recursively, using the previous bounds on $\eta_{<i}$.

2.19 Approximately Unbiased Simulator (AUS)

Assume B compact. The Gibbs Sampler Simulator for H_C is only approximately unbiased, although convergence is at a geometric rate. The GHK simulator (42) for H_C was also shown to have geometric convergence. Using the method in Lemma 4 for simulating $1/P$, it is also possible to use the NIS, SDS, PCF, or DCS simulators to get smooth, approximately unbiased simulators of H_C with the property that the bias converges to zero at a geometric rate in the number of repetitions τ . The KFS with a positive kernel, such as the normal or logistic, satisfies Lemma 6 for all $q > 0$. Taking ω proportional to $e^{-\alpha\tau}$ for some $\alpha > 0$ then implies that Lemma 4 can be used for this simulator, with the bias introduced by kernel-smoothing contributing an additional bias term of order $e^{-\alpha\tau q/(1+q)}$ to the geometric bias coming from the truncation at τ in Lemma 4. Again, a geometric rate is achieved overall.

In principle, the method in Lemma 5 could also be used to construct approximately unbiased simulators of H_C . This method might be better than the method in Lemma 4 for small τ and some simulators, but for large τ the geometric rate achieved in Lemma 4 will dominate the $1/\tau$ rate achieved in Lemma 5.

3 Test Problems and Computational Results

The Appendix describes a series of GAUSS and FORTRAN procedures for simulation of multivariate normal rectangle probabilities and the derivatives of these probabilities with respect to the mean and covariances of the normal distribution.¹⁶ In this Section, we describe the test problems we used to evaluate the operational characteristics of these algorithms and then discuss the findings from our computational experiments.

3.1 Description of the Test Problems

A case that yields multivariate normal rectangle probabilities that are easily calculated analytically or by quadrature is the one-factor model,

$$(67) \quad V = \mu + S\eta + \Lambda\epsilon,$$

where S is an $m \times m$ diagonal matrix with diagonal elements s_i , Λ is an $m \times 1$ array of factor loadings, μ is an $m \times 1$ vector of means, η is an $m \times 1$ vector of independent standard normal variates, and ϵ is an independent $m \times 1$ standard normal vector. Given ϵ , the constraints require

$$(68) \quad S^{-1}(a - \mu - \Lambda\epsilon) < \eta < S^{-1}(b - \mu - \Lambda\epsilon).$$

Define

$$(69) \quad Q_i = \Phi((b_i - \mu_i - \lambda_i\epsilon)/s_i) - \Phi((a_i - \mu_i - \lambda_i\epsilon)/s_i)$$

¹⁶Both versions of the programs are available from the authors upon request.

and

$$(70) \quad q_i = \phi((b_i - \mu_i - \lambda_i \epsilon)/s_i) - \phi((a_i - \mu_i - \lambda_i \epsilon)/s_i).$$

Then,

$$(71) \quad P = \int_{-\infty}^{+\infty} \left(\prod_{i=1}^m Q_i \right) \phi(\epsilon) d\epsilon,$$

$$(72) \quad \partial P / \partial \mu_m = (-1/s_m) \int_{-\infty}^{+\infty} \left(\prod_{i=1}^{m-1} Q_i \right) q_m \phi(\epsilon) d\epsilon,$$

and

$$(73) \quad \partial P / \partial \lambda_m = (-1/s_m) \int_{-\infty}^{+\infty} \left(\prod_{i=1}^{m-1} Q_i \right) q_m \epsilon \phi(\epsilon) d\epsilon.$$

These expressions can be evaluated by one-dimensional Gaussian quadrature, and in a few cases evaluated analytically, which will provide benchmarks to gauge the accuracy of the 13 simulation algorithms in the experiments below. One analytic case occurs when the factor is loaded only on the last alternative, so that it is equivalent to a change in the scale of V_m : transform s_m to $(s_m^2 + \lambda_m^2)^{1/2}$ and λ_m to zero, so that the terms Q_i and q_m are independent of ϵ , and Q_m and q_m are evaluated at the transformed values,¹⁷ obtaining

$$(74) \quad P = \left(\prod_{i=1}^m Q_i \right),$$

$$\partial P / \partial \mu_m = - \left(\prod_{i=1}^{m-1} Q_i \right) q_m \cdot (s_m^2 + \lambda_m^2)^{-1/2},$$

$$\partial P / \partial \lambda_m = - \left(\prod_{i=1}^{m-1} Q_i \right) q_m \{ (b_m - \mu_m) - (a_m - \mu_m) \} \lambda_m \cdot (s_m^2 + \lambda_m^2)^{-3/2}.$$

A second analytic case occurs when $a = -\infty$, $b = 0$, $\mu = 0$, S is the identity matrix, and Λ is a vector of ones. Then,

$$(75) \quad P = \int_{-\infty}^{+\infty} \Phi(-\epsilon)^m \phi(-\epsilon) d\epsilon = 1/(m+1),$$

$$\partial P / \partial \mu_m = \int_{-\infty}^{+\infty} \Phi(-\epsilon)^{m-1} \phi(-\epsilon)^2 d\epsilon,$$

$$\partial P / \partial \lambda_m = \int_{-\infty}^{+\infty} \Phi(-\epsilon)^{m-1} \phi(-\epsilon)^2 \epsilon d\epsilon.$$

¹⁷The convolution $Pr(V_m < v_m) = \int_{-\infty}^{+\infty} \Phi((v_m - \mu_m - \lambda_m \epsilon)/s_m) \phi(\epsilon) d\epsilon = \Phi\left(\frac{v_m - \mu_m}{\sqrt{s_m^2 + \lambda_m^2}}\right)$ is used to obtain the analytic form.

Finally, we analyzed the case of a random effect combined with an autoregressive structure of order one, for which no analytic solution exists. This model is given by

$$(76) \quad V_{ni} = \epsilon_n + \eta_{ni}, \quad \eta_{ni} = \rho\eta_{n,i-1} + \nu_{ni}, \quad i = 1, \dots, m,$$

where ϵ and ν are mutually independent, the random effect ϵ_n is *i.i.d.* over n , and the AR(1) innovation ν_{ni} is *i.i.d.* over both n and i . Since the evaluation of the probability, linear derivative, and logarithmic derivative expressions in this case requires numerical integration of the order of the dimension m , it was not tractable computationally to use numerical quadrature as a benchmark. Instead we chose in this case to obtain benchmark results by averaging a very large number of simulations (20,000) of the *GHK* method, which as it will become clear below, seems overall to be the most reliable procedure throughout the cases studied.

3.2 Comparative Performance

For each computational experiment we used five hundred Monte Carlo repetitions of all thirteen simulation algorithms. The number of simulations in calculating empirical expectations of the H matrix function was chosen endogenously by our programs, so as to require approximately the same time for each simulation method. The specific results reported in this Section were obtained through the GAUSS implementation of the routines on 386/16MHz Personal Computers.

Figures 2–6 describe the first series of experiments, in which truncated normal vectors V of dimension $m = 2$ were generated, having the factor structure (67). Figure 2 gives the 6 types of correlation/covariance structure¹⁸ we studied, with $\omega_1 = 1$, $\omega_2 = \{1 \text{ or } 8\}$, and $\rho_{12} = \{0, .6, \text{ or } .9\}$, where $\text{var } V_i \equiv \omega_i$ and $\text{cov}(V_1, V_2) \equiv \rho_{12} \cdot \omega_1 \cdot \omega_2$. Figure 3 describes the 14 different rectangles/restrictions¹⁹ we investigated. These rectangles were chosen so as to analyze the effect of symmetry around either or both axes, as well as the location of them either close to the center of the distribution or far out in the tails. Hence, the results summarized in Figures 4–6 refer to the 84 cases, $\{A1, A2, \dots, N5, N6\}$, that are obtained by combining these 6 correlation structures with the 14 sets of restrictions. Table 2a summarizes the characteristics of these 84 experiments in terms of the exact probability and the condition number and determinant of the variance-covariance matrix Ω in each case. In Table 2b we give the explicit loadings in terms of the factor structure (67) for the six variance-covariance structures used in the experiments, i.e., $S \equiv \begin{pmatrix} s_1 & 0 \\ 0 & s_2 \end{pmatrix}$ and $\Lambda \equiv \begin{pmatrix} \lambda_1 \\ \lambda_2 \end{pmatrix}$.

For each one of the 84 cases studied, we rated the methods in terms of root-mean-squared-error relative to the best method for that case, e.g., a *RMSE* Rating of 0.5 means that the method in question exhibited double the *RMSE* of the method with the lowest *RMSE* for that case.

¹⁸indexed by a number from 1 to 6

¹⁹indexed by a letter from A to N

**Table 2a: Characteristics of 84 Experiments
Restrictions {A-N} × Variance-Covariances {1-6}
Exact Probability of Each Case***

	V.-Cov.1	V.-Cov.2	V.-Cov.3	V.-Cov.4	V.-Cov.5	V.-Cov.6
Restr. A (3 × 2)	0.01133	0.02113	0.02143	0.04625	0.05656	0.07206
Restr. B (3 × 2)	0.34042	0.34112	0.34135	0.04960	0.04973	0.04974
Restr. C (3 × 2)	0.01133	0.00095	8.601e-08	0.04625	0.03617	0.02069
Restr. D (3 × 2)	4.101e-05	0.00081	0.00174	0.00017	3.674e-05	9.799e-11
Restr. E (3 × 2)	0.00092	0.00023	2.662e-08	0.00013	1.366e-05	1.118e-11
Restr. F (3 × 2)	4.101e-05	1.036e-09	1.123e-13	0.00017	4.088e-06	1.633e-12
Restr. G (5 × 5)	0.12977	0.15259	0.15422	0.03720	0.03134	0.00792
Restr. H (5 × 5)	0.12977	0.08309	0.04057	0.03720	0.01960	0.00129
Restr. I (10 × 2)	0.02272	0.02272	0.02272	0.09276	0.09276	0.09276
Restr. J (10 × 2)	0.68269	0.68269	0.68269	0.09948	0.09948	0.09948
Restr. K (10 × 2)	0.02272	0.02272	0.02272	0.09276	0.09276	0.09276
Restr. L (6 × 4)	0.00135	0.00121	0.00074	0.16260	0.16295	0.16304
Restr. M (6 × 4)	0.95192	0.95311	0.95444	0.19688	0.19738	0.19741
Restr. N (6 × 4)	0.00135	0.00121	0.00074	0.16260	0.16295	0.16304
Cond.Num.(Ω)	1.000	3.999	18.999	63.999	101.139	345.446
Det.(Ω)	1.000	0.640	0.190	64.000	40.960	12.160

* After the name of each restriction type, the dimensions of the rectangle appear in parentheses.

Analytically,²⁰

$$\text{Average RMSE Rating of Method } j \equiv \frac{1}{84} \sum_{k=1}^{84} \text{RMSE}(\text{best method for case } k) / \text{RMSE}(\text{method } j \text{ in case } k)$$

In each figure the results are presented in four subfigures, (a) to (d). The first two report performance ratings for each of the 14 restriction types, averaging across all 6 correlation structures, with subfigures (a) and (b) giving cases A–G and H–N respectively. Subfigure (c) reports performance for each of the 6 types of correlation, averaging across the 14 restriction types. Finally, in subfigure (d) we report overall performance, averaging across all 84 correlation/restriction cases. The results of simulating probabilities appear in Figure 4, those for simulating linear derivatives in Figure 5, and those for logarithmic derivatives in Figure 6. Overall the 84 cases studied, the average ratings of the various methods are also summarized in Table 3.

Our results support the following conclusions: *GHK* can be recommended as unambiguously the most reliable method for simulating normal rectangle probabilities, achieving an overall rating of 93%, compared to only 52% for the next best method, *NIST*.²¹ For simulating derivative expressions, the *PCF* method exhibited the highest overall rating, 79% for linear and 79% for logarithmic, while *NISE* was second and *GHK* third for both types of derivatives. It is interesting

²⁰Interested readers may request from the authors considerably more detailed tables that report bias, variance, mean-squared-error, quantiles, robust statistics, and timing results for all cases studied.

²¹Indeed, *GHK* achieved a first-place rating in over 70 of the 84 cases studied.

**Table 2b: Characteristics of 84 Experiments
Factor Structures of Variance-Covariances {1-6}**

	$\Omega(1)$	$\Omega(2)$	$\Omega(3)$	$\Omega(4)$	$\Omega(5)$	$\Omega(6)$
s_1	0.0	0.0	0.0	0.0	0.0	0.0
s_2	1.0	0.8	0.43589	8.0	6.4	3.48712
λ_1	1.0	1.0	1.0	1.0	1.0	1.0
λ_2	0.0	0.6	0.9	0.0	4.8	7.2
ω_1^2	1.0	1.0	1.0	1.0	1.0	1.0
$\omega_2^2 = s_2^2 + \lambda_2^2$	1.0	1.0	1.0	64.0	64.0	64.0
$\rho_{12} = \lambda/\omega_2$	0.0	0.6	0.9	0.0	0.6	0.9

Table 3: Average RMSE Ratings* across 84 Cases

Probabilities		Linear Derivatives		Logarithmic Derivatives	
GHK	0.928	PCF	0.786	PCF	0.792
NIST	0.523	NISE	0.597	NISE	0.577
NISE	0.450	GHK	0.582	GHK	0.557
PCF	0.291	NIST	0.559	NIST	0.536
DCS	0.163	SDS	0.290	SDS	0.340
CFS	0.145	CFS	0.282	CFS	0.268
SDS	0.143	DCS	0.211	KFS	0.221
KFS	0.025	KFS	0.079	DCS	0.190
AUS	**	AUS	**	GSS	0.136
ARSE	**	ARSE	**	ARSR	0.123
ARSR	**	ARSR	**	SUS	0.101
GSS	**	GSS	**	AUS	0.100
SUS	**	SUS	**	ARSE	0.022

* Average RMSE Rating of method $j \equiv \frac{1}{84} \sum_{k=1}^{84} RMSE(\text{best method in case } k) / RMSE(\text{method } j \text{ for case } k)$

** AUS, ARSE, ARSR, GSS, and SUS not applicable

to note that the normal importance sampling methods, *NISE* and *NIST*, seem to perform fairly well, bettered only by *GHK* in the case of probabilities, and by one (*PCF*) or two (*PCF* and *GHK*) methods in the case of derivatives. What Figures 4–6 illustrate more clearly than the overall ratings table above is that *GHK* appears more robust than all other methods, in that it performs at or near the top in each one of the 84 cases studied. In particular, it performs even more impressively relative to the other algorithms in the most difficult cases of either high correlation among the elements of V and/or very low probability mass in the restriction region.

In the next set of computational experiments we investigated the consequences for the performance of the simulation methods of increasing the number of simulations used. We selected 6 cases out of the 84 possible ones discussed above²² in such a way as to provide a wide range for the performance of the preferred method, *GHK*. The results are summarized in Figure 7, where the actual *RMSE* of each of the 5 best methods, averaged over these 6 cases, is plotted against the

²²A5, G6, K1, K2, L3, and L4

time required to perform $\times 1$, $\times 2$, $\times 3$, $\times 4$, and $\times 5$ the original number of simulations used. The results again highlight the superior performance of *GHK*, especially for simulating probabilities. For linear derivatives, *PCF* has a slight edge over *GHK*, while for logarithmic derivatives, there is virtually a tie between *GHK* and *PCF*.

In the final set of experiments, we analyzed the impact on *RMSE* of increasing the dimension m of the problem. Figures 8–9 report results for the 5 best methods for $m = \{2, 4, 8, \text{ or } 16\}$. In Figure 8, the $m \times 1$ multivariate normal vector V has the *one-factor* structure (*RE1*)

$$(75') \quad V_{ni} = \epsilon_n + \nu_{ni},$$

with $\sigma_\nu = 1$ and $\sigma_\epsilon = 1$. In Figure 9, the *one-factor plus AR(1)* structure (*RE1AR1*) was used

$$(76') \quad V_{ni} = \epsilon_n + \eta_{ni}, \quad \eta_{ni} = \rho\eta_{n,i-1} + \nu_{ni}, \quad i = 1, \dots, m,$$

with $\sigma_\nu = 0.6$, $\sigma_\epsilon = 1$, and $\rho = 0.8$. The parameters were chosen so as to generate cases for which *GHK* is not the dominating method. In particular, the restriction region in these experiments was the all-negative orthant, which with the addition of a very high degree of serial correlation implies that the probability of being in the restricted region is not very small.²³ Figure 8 shows that *GHK* remains the only reliable method when the dimension is increased: the performance of *GHK* is uniformly superior, whether one simulates probabilities, linear derivatives, or logarithmic derivatives. The other methods perform especially poorly as m rises. Finally, Figure 9 shows the effects of increasing the serial correlation through the addition of the *AR(1)* structure. The methods least affected by increasing m are *GHK* and *CFS*, with *GHK* once again exhibiting the most satisfactory performance.²⁴

It should be noted that the results presented here discriminate against methods, like *GSS*, that are *not vectorizable*, since *GAUSS* is particularly efficient for vector operations. In preliminary timings using our FORTRAN code, we confirmed that methods that are difficult to vectorize then gain in relative speed. The impact of vectorization techniques on the performance of the simulation algorithms we studied is potentially a very important issue that needs to be investigated further. We plan future work on this topic.

4 Conclusions

The problem of evaluating multivariate normal probabilities and their derivatives is an important one in econometrics and biometrics because such expressions appear in leading econometric models, such as the *multinomial probit* (MNP) and other limited dependent variable models based on normality. Estimation of these models requires, for each trial parameter vector and each observation

²³The true probability was 0.3333 for $m = 2$, 0.2001 for $m = 4$, 0.1095 for $m = 8$, and 0.0565 for $m = 16$ in the *RE1* case, and 0.3974, 0.2855, 0.1753, and 0.0939 respectively for the *RE1AR1* case.

²⁴In view of the relatively high probabilities in this last set of experiments, it is not surprising that *CFS* performs well in them.

in a sample, evaluation of such probability expressions and their derivatives. The problem is computationally difficult unless the dimension of the integral is less than four or the covariance matrix Ω has a special structure, such as a factorial structure with a low number of factors.

This paper surveyed Monte Carlo techniques that have been developed for approximating normal orthant probabilities and their derivatives that limit computation while possessing properties that facilitate their use in iterative calculations for statistical inference. We considered the following methods: the Crude Frequency Simulator (CFS), two variants of Normal Importance Sampling (NISE and NIST), a Kernel-Smoothed Frequency Simulator (KFS), Stern's Decomposition Simulator (SDS), the Geweke-Hajivassiliou-Keane Simulator (GHK), a Parabolic Cylinder Function Simulator (PCF), Deák's Chi-squared Simulator (DCS), two variants of Acceptance/Rejection Simulation (ARSE and ARST), the Gibbs Sampler Simulator (GSS), a Sequentially Unbiased Simulator (SUS), and an Approximately Unbiased Simulator (AUS). In an Appendix we described GAUSS and FORTRAN algorithms implementing these methods. We presented test problems we used to investigate the operational properties of the methods, focussing on *RMSE* rankings, and summarized our computational experience with them. We also examined the impact of increasing the number of simulations R and the dimension of the probability integral m . We concluded that the *GHK* simulator appears overall the most reliable method, especially for simulating orthant probabilities.

REFERENCES

1. Abramowitz, M. and I. Stegun (1964). *Handbook of Mathematical Functions*. Washington: National Bureau of Standards.
2. Börsch-Supan, A. and V. Hajivassiliou (1993). "Smooth Unbiased Multivariate Probability Simulators for Maximum Likelihood Estimation of Limited Dependent Variable Models," *Journal of Econometrics*, **58**(3), 347-368.
3. Clark, C. (1961). "The Greatest of a Finite Set of Random Variables," *Operations Research* **9**, 145-162.
4. Daganzo, C. (1980). *Multinomial Probit*. New York: Academic Press.
5. Davis, P. and P. Rabinowitz (1984). *Methods of Numerical Integration*. New York: Academic Press.
6. Deák, I. (1980a). "Three Digit Accurate Multiple Normal Probabilities," *Numerische Mathematik* **35**, 369-380.
7. Deák, I. (1980b). "Fast Procedures for Generating Stationary Normal Vectors," *Journal of Statistical Computation and Simulation* **10**, 225-242.
8. Devroye, L. (1986). *Non-Uniform Random Variate Generation*. New York: Springer.
9. Dutt, J. (1973). "A Representation of Multivariate Normal Probability Integrals by Integral Transforms," *Biometrika* **60**, 637-645.
10. Dutt, J. (1976). "Numerical Aspects of Multivariate Normal Probabilities in Econometric Models," *Annals of Economic and Social Measurement* **5**, 547-562.
11. Fishman, G. (1973). *Concepts and Methods of Digital Simulation*. New York: Wiley.
12. Geman, S. and D. Geman (1984). "Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **6**, 721-741.
13. Geweke, J. (1989). "Efficient Simulation from the Multivariate Normal Distribution Subject to Linear Inequality Constraints and the Evaluation of Constraint Probabilities," mimeo, Duke University.
14. Hajivassiliou, V. (1993). "Simulation Estimation Methods for Limited Dependent Variable Models," in *Handbook of Statistics, Vol.11 (Econometrics)*, G.S. Maddala, C.R. Rao, and H.D. Vinod (eds.). Amsterdam: North-Holland.
15. Hajivassiliou, V. and D. McFadden (1990). "The Method of Simulated Scores for the Estimation of LDV Models," Cowles Foundation Discussion Paper No.967, Yale University.
16. Hammersley, J. and D. Handscomb (1964). *Monte Carlo Methods*. London: Methuen.
17. Horowitz, J., J. Sparmann, and C. Daganzo (1981). "An Investigation of the Accuracy of the Clark Approximation for the Multinomial Probit Model," *Transportation Science* **16**, 382-401.

18. Keane, M. (1994). "A Computationally Efficient Practical Simulation Estimator for Panel Data, with Applications to Estimating Temporal Dependence in Employment and Wages," *Econometrica*, **62**(1), 95-116.
19. Lerman, S. and C. Manski (1981). "On the Use of Simulated Frequencies to Approximate Choice Probabilities," in C. Manski and D. McFadden (ed.), *Structural Analysis of Discrete Data with Econometric Applications*. Cambridge: MIT Press, pp. 305-319.
20. McFadden, D. (1984). "Econometric Analysis of Qualitative Response Models," in Z. Griliches and M. Intriligator (ed.), *Handbook of Econometrics*, Vol. 2. Amsterdam: North Holland, pp. 1395-1457.
21. McFadden, D. (1989). "A Method of Simulated Moments for Estimation of Discrete Response Models without Numerical Integration," *Econometrica*, **57**, 995-1026.
22. McFadden, D. and P. Ruud (1990). "Estimation by Simulation," M.I.T. Working Paper.
23. Moran, P. (1984). "The Monte Carlo Evaluation of Orthant Probabilities for Multivariate Normal Distributions," *Australian Journal of Statistics* **26**, 39-44.
24. Owen, D. (1956). "Tables for Computing Bivariate Normal Probabilities," *Annals of Mathematical Statistics* **27**, 1075-1090.
25. Pakes, A. and D. Pollard (1989). "Simulation and the Asymptotics of Optimization Estimators," *Econometrica* **57**, 1027-1057.
26. Quandt, R. (1984). "Computational Problems in Econometrics," in Z. Griliches and M. Intriligator (eds.), *Handbook of Econometrics*, Vol. 1. Amsterdam: North Holland, pp. 1395-1457.
27. Rubinstein, R. (1981). *Simulation and the Monte Carlo Method*. New York: Wiley.
28. Ruud, P. (1986). "On the Method of Simulated Moments for the Estimation of Limited Dependent Variable Models," mimeo, University of California at Berkeley.
29. Ruud, P. (1991). "Extensions of Estimation Methods Using the EM Algorithm," *Journal of Econometrics*, **49**, 305-341.
30. Siegmund, D. (1985). *Sequential Analysis*. New York: Springer-Verlag.
31. Spanier, J. and K. Oldham (1987). *An Atlas of Functions*. Washington: Hemisphere.
32. Stern, S. (1992). A Method for Smoothing Simulated Moments of Discrete Probabilities in Multinomial Probit Models. *Econometrica* **60**:943-952.
33. Stroud, A. (1971). *Approximate Calculation of Multiple Integrals*. New York: Prentice Hall.
34. Thisted, R. (1988). *Elements of Statistical Computing*. Chapman-Hall.

Appendix

Description of GAUSS and FORTRAN Procedures

The simulation methods presented in this paper have been coded in GAUSS and in FORTRAN. Both versions of the programs are available via anonymous FTP from the Internet site econ.yale.edu, subdirectory pub/vassilis/simulation. Each simulator procedure requires the following standard inputs; the interpretation of some inputs may vary from routine to routine, and not all are used in all routines:

<i>M</i>	Dimension of the multivariate normal
<i>VMU</i>	Mean of multivariate normal, an $M \times 1$ vector
<i>W</i>	Covariance matrix of multivariate normal, an $M \times M$ array
<i>WI</i>	Inverse of <i>W</i>
<i>C</i>	Lower triangular Choleski factor of <i>W</i> , an $M \times M$ array
<i>A</i>	Lower bound of rectangle, an $M \times 1$ vector. {When the lower bound is $-\infty$, set $A = (-1.0E10)*ONES(M, 1)$.}
<i>B</i>	Upper bound of rectangle, an $M \times 1$ vector. {When the upper bound is ∞ , set $B = (1.0E10)*ONES(M, 1)$.}
<i>NR</i>	Number of repetitions
<i>U</i>	Random variates, an $M \times R$ array
<i>PARAM</i>	Parameters and constants for the simulation

The simulators all return $\{P, HU, HC\}$, where *P* is the scalar rectangle probability, *HU* is the $(M + 1) \times (M + 1)$ array of unconditional partial moments (6), and *HC* is the $(M + 1) \times (M + 1)$ array of conditional moments (7). Parts of the output not provided by a simulator are set to -999.

In the FORTRAN implementation, two additional inputs are required, *MMAX* and *NRMAX*, specifying the maximum values of *M* and *R* allocated at compilation time.

The programs include code for all statistical functions, spherical transformations, and antithetics routines that are required by the simulation algorithms, and hence are self-contained.²⁵

²⁵In the Monte Carlo experiments reported in this paper, the number 1753227 was used as the starting seed for generating the random variates.

Figure 2
Correlation/Covariance Structures Studied

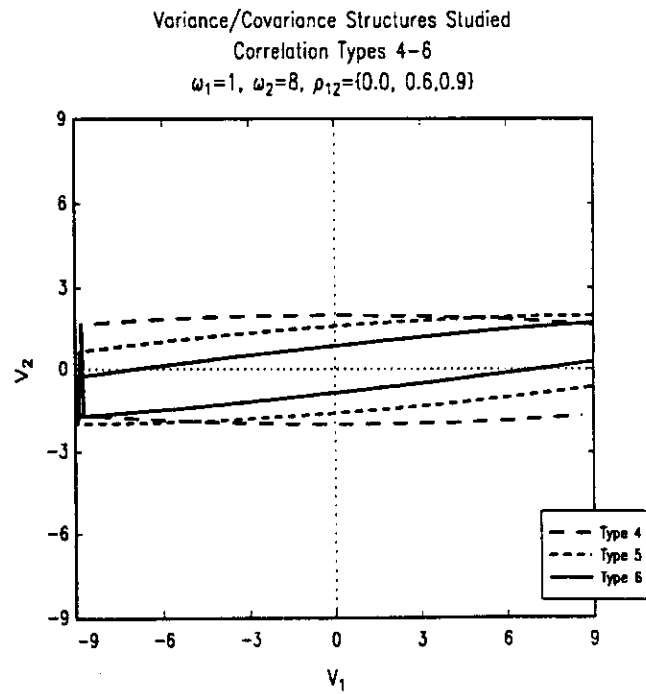
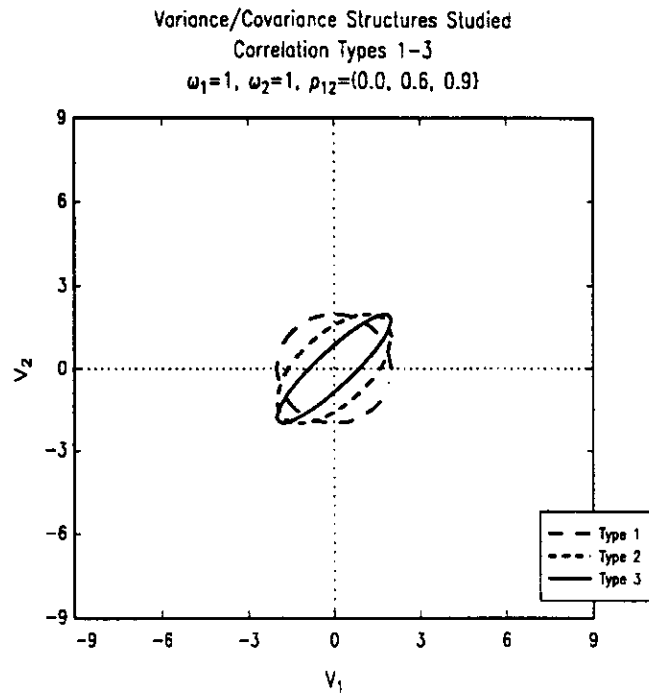


Figure 3
Rectangles/Restrictions Studied

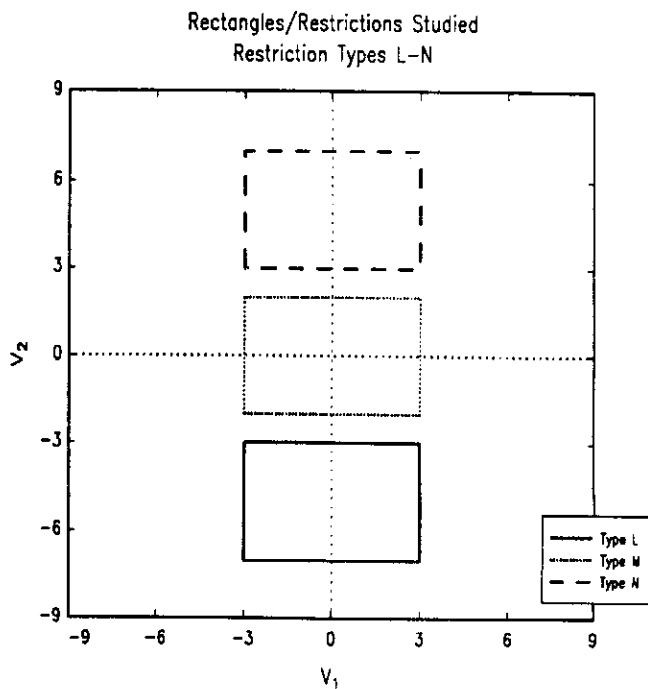
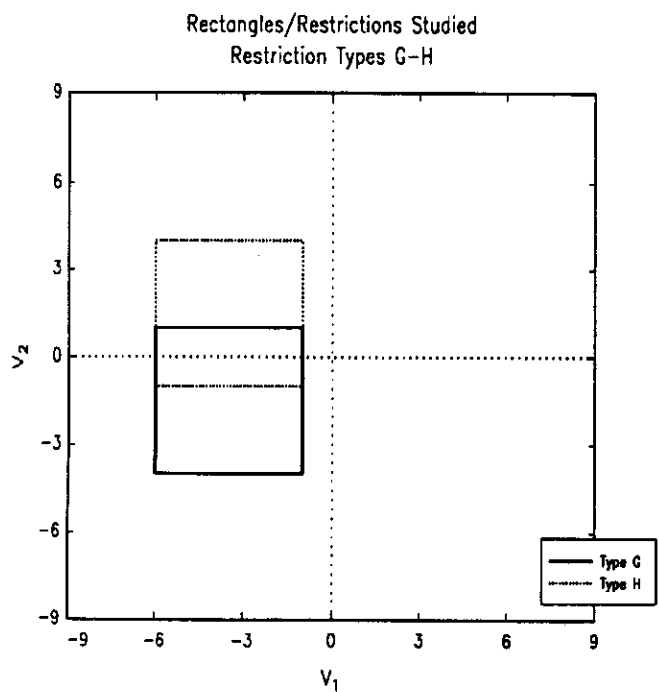
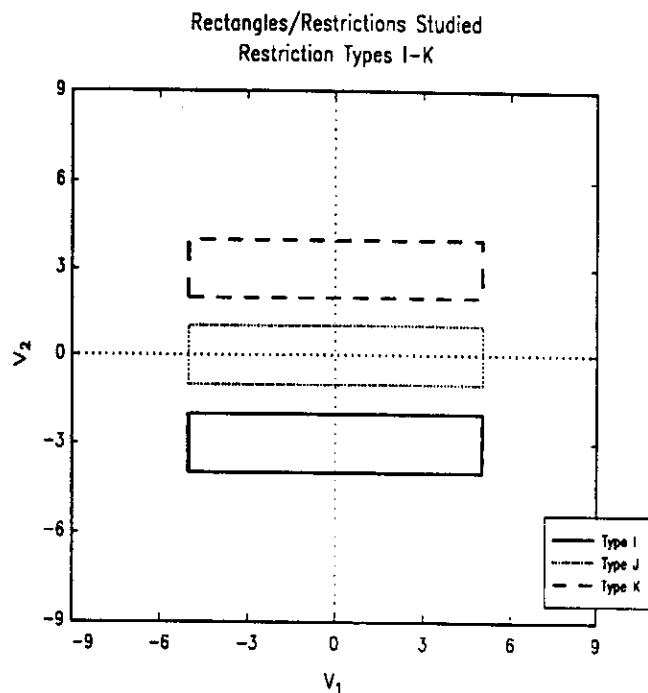
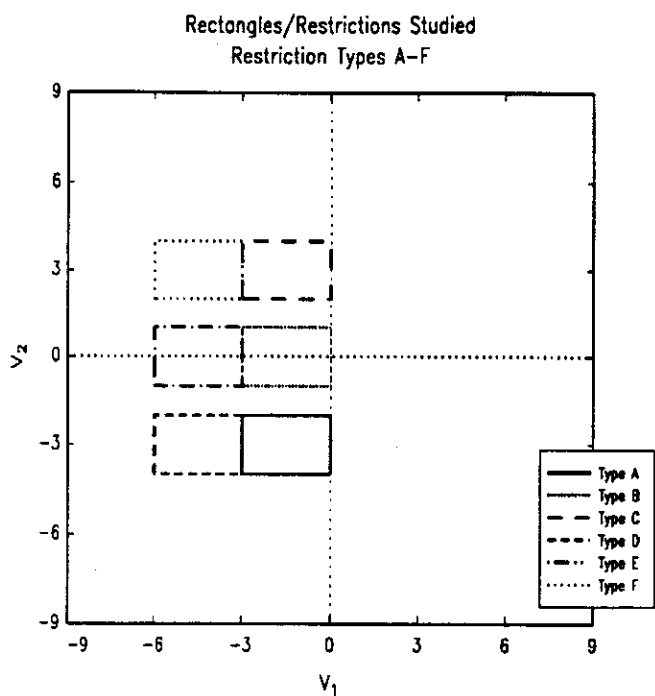


Figure 4

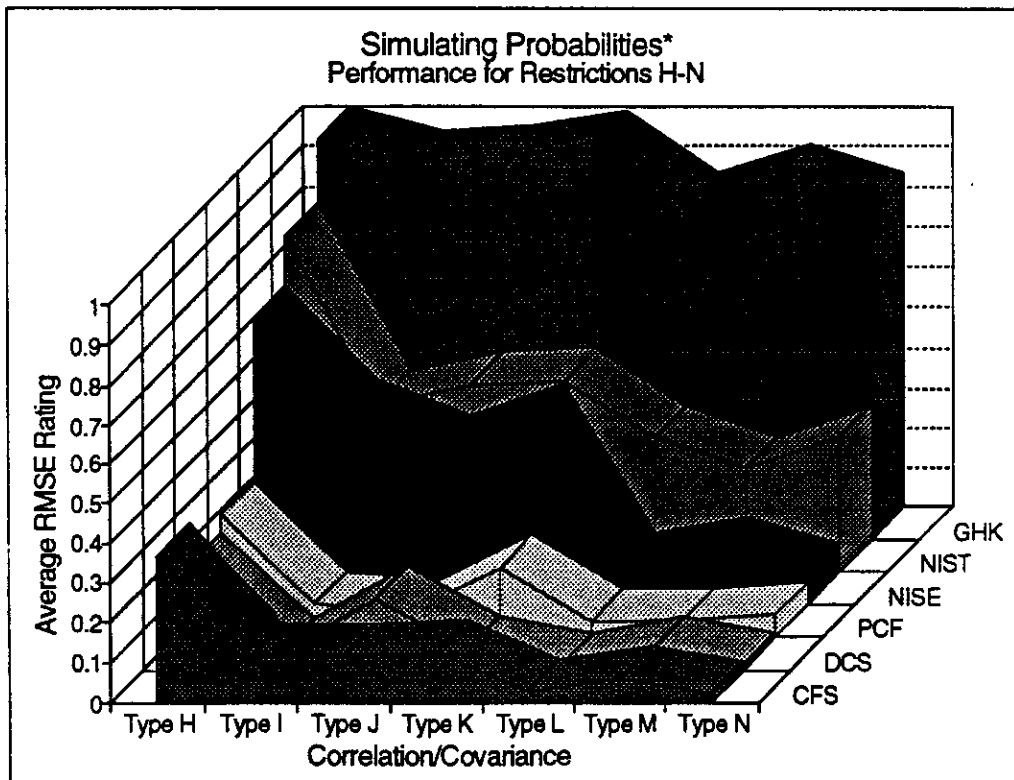
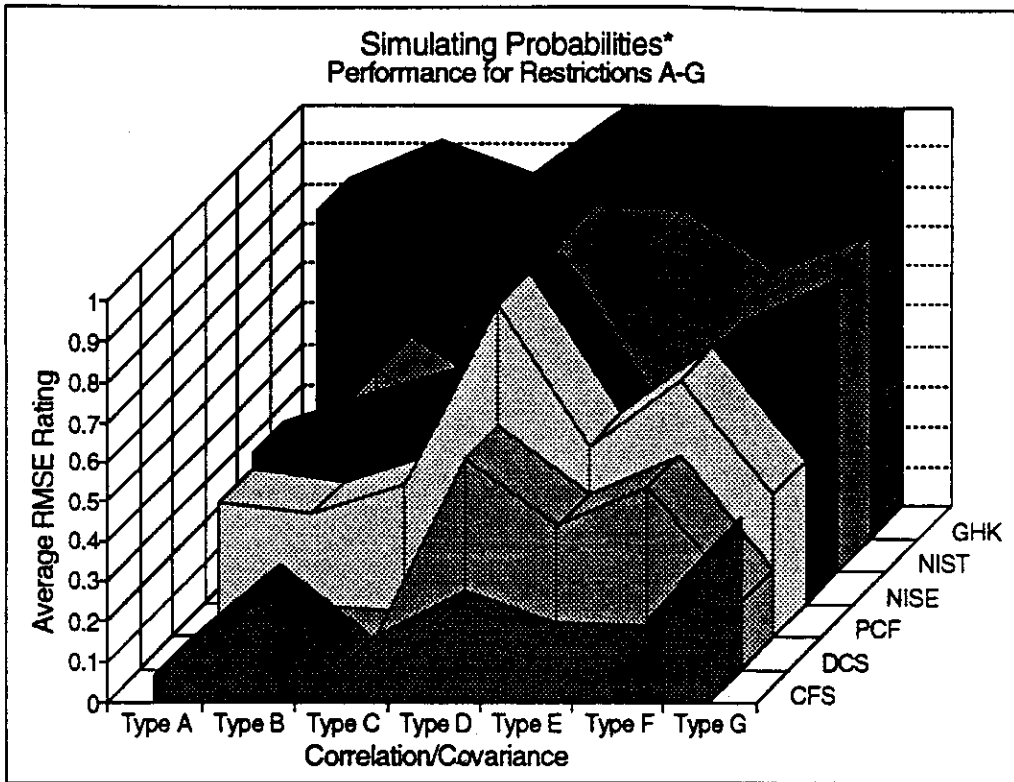


Figure 4 (continued)

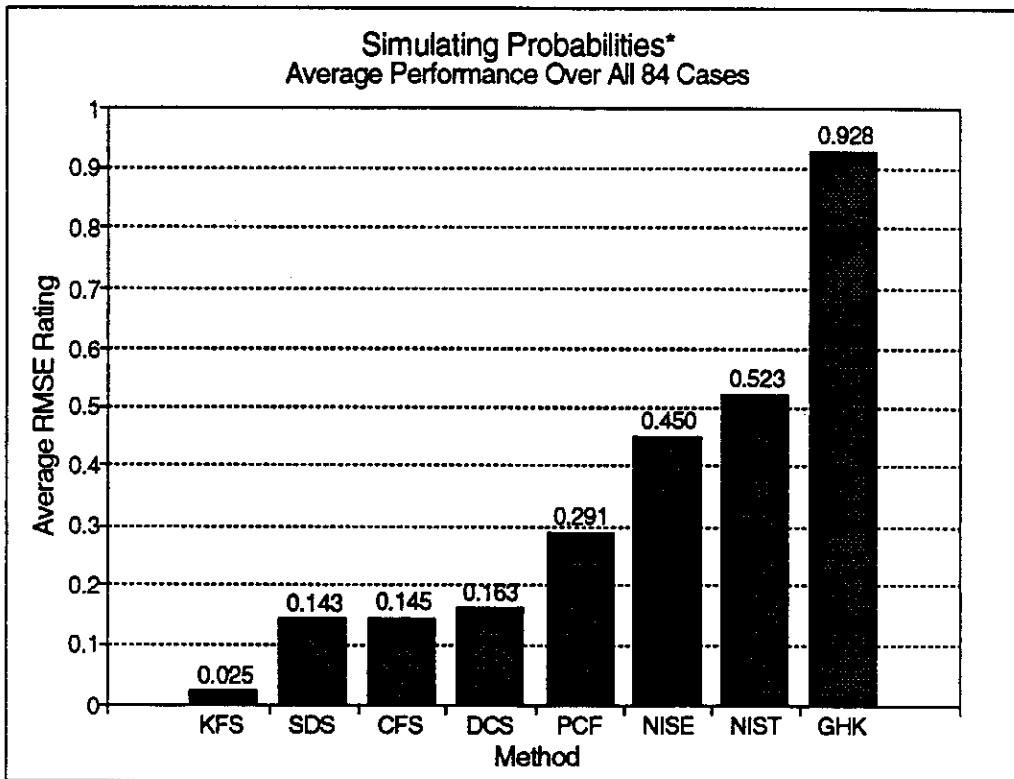
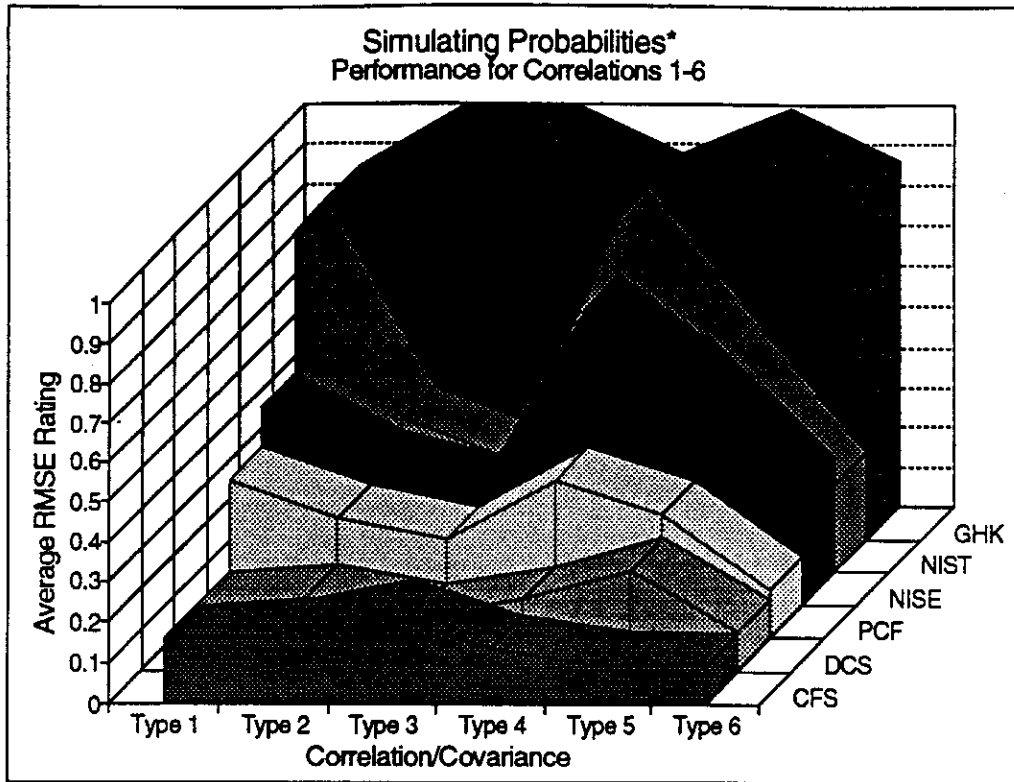


Figure 5

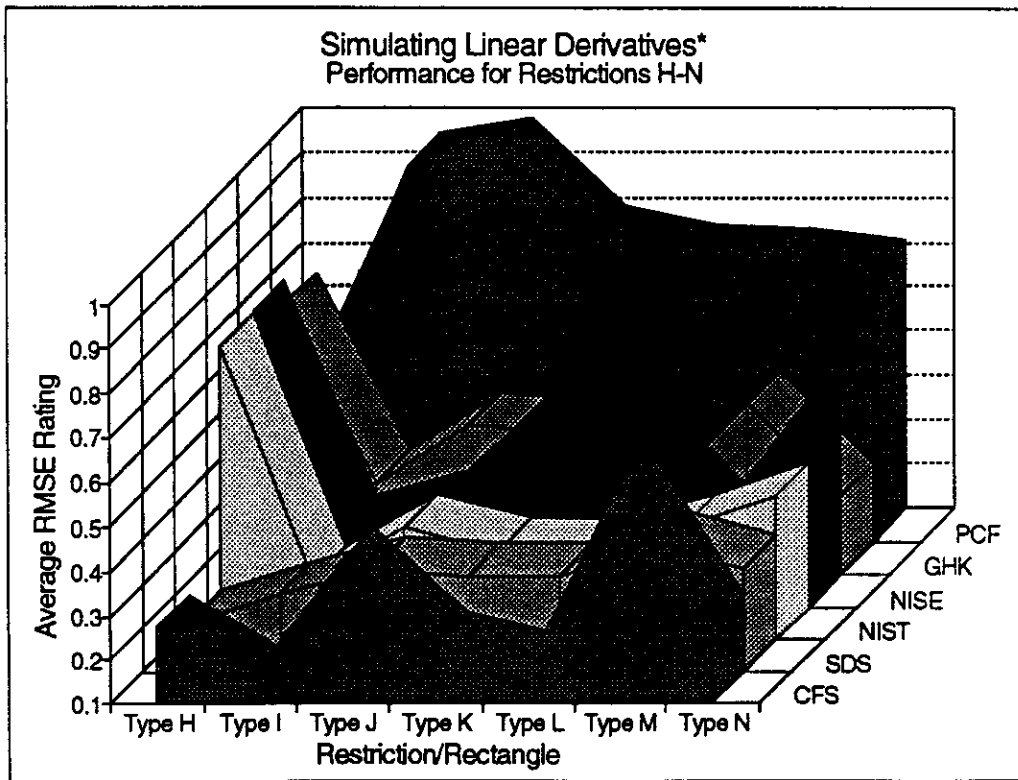
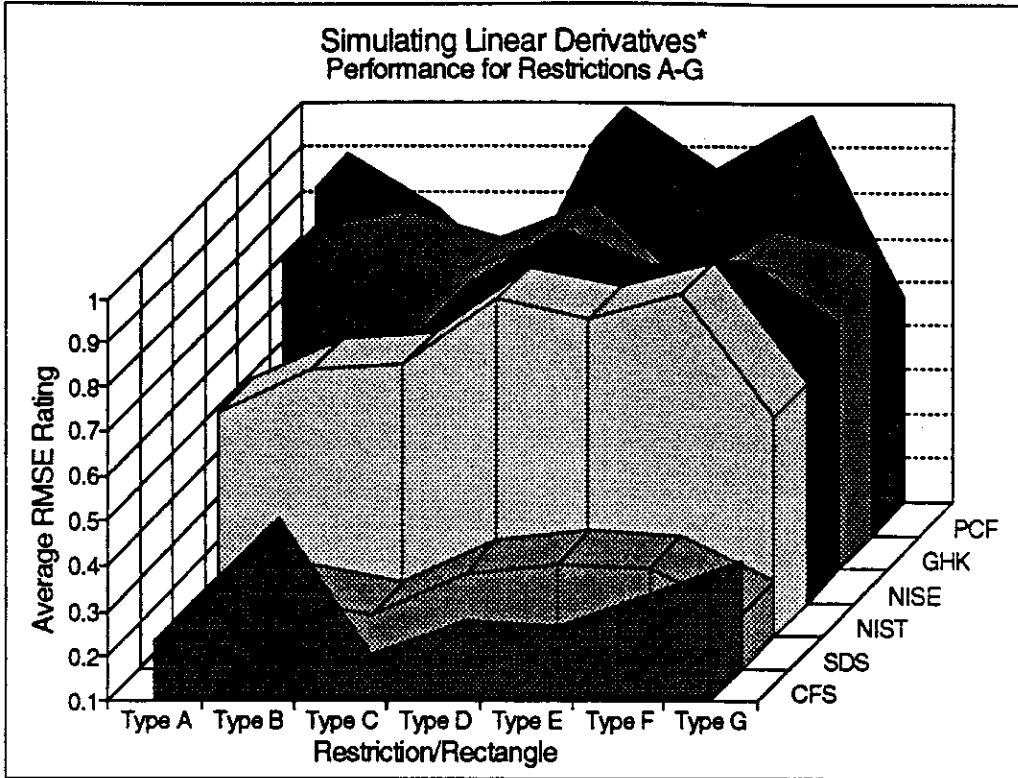


Figure 5 (continued)

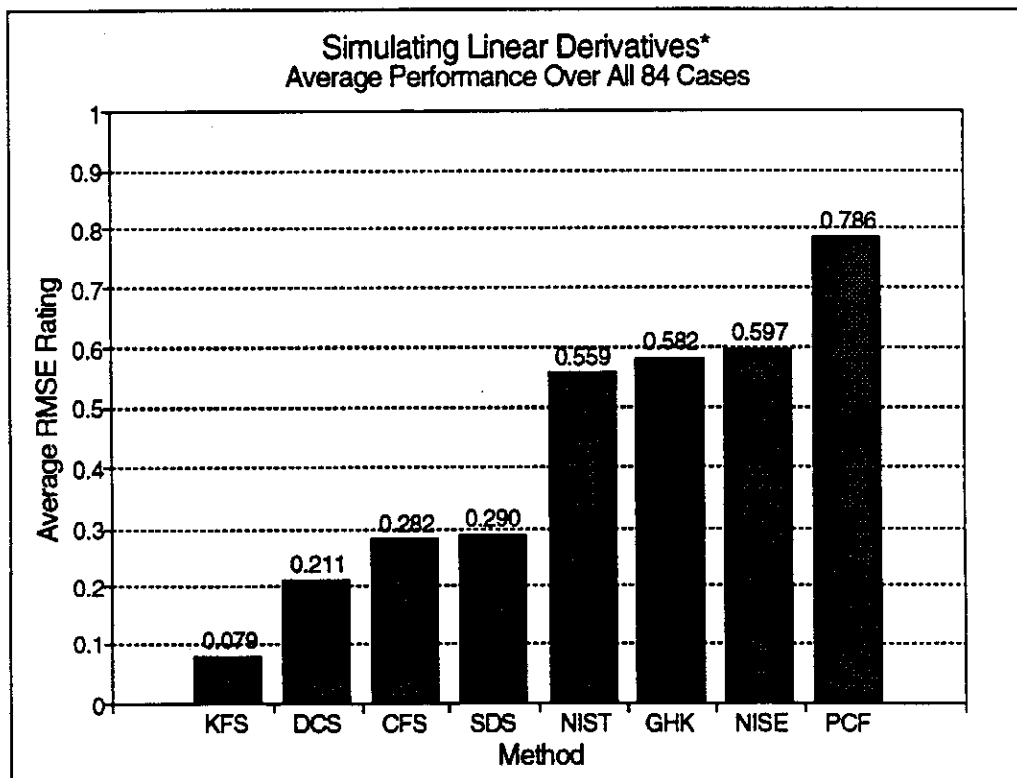
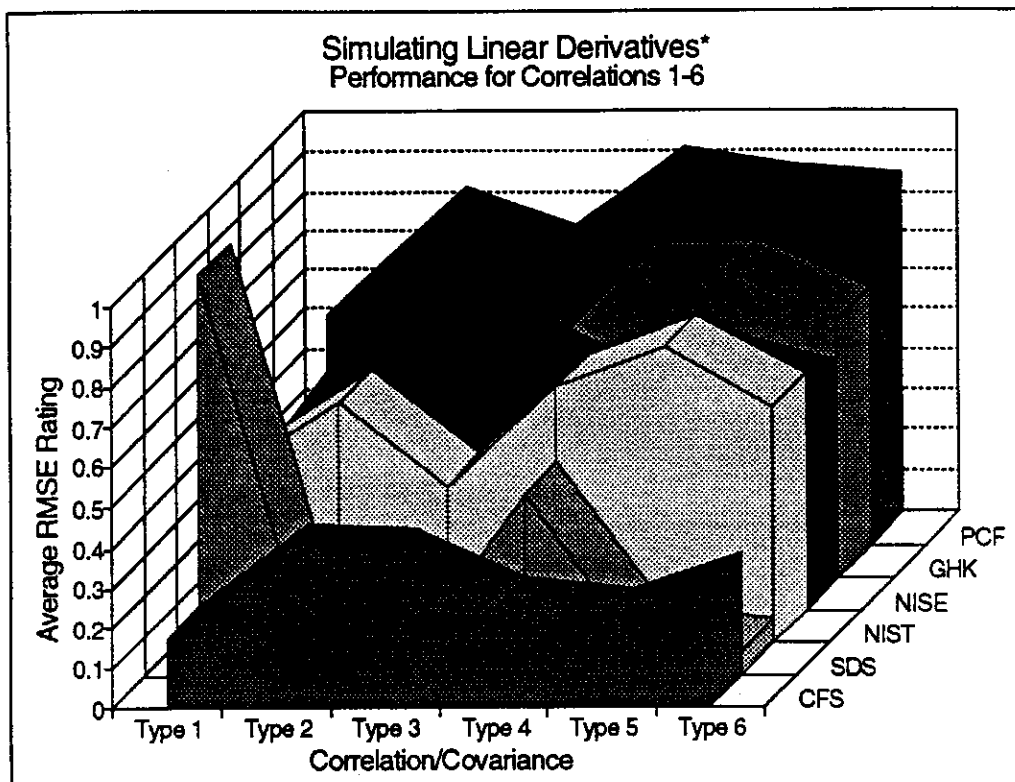


Figure 6

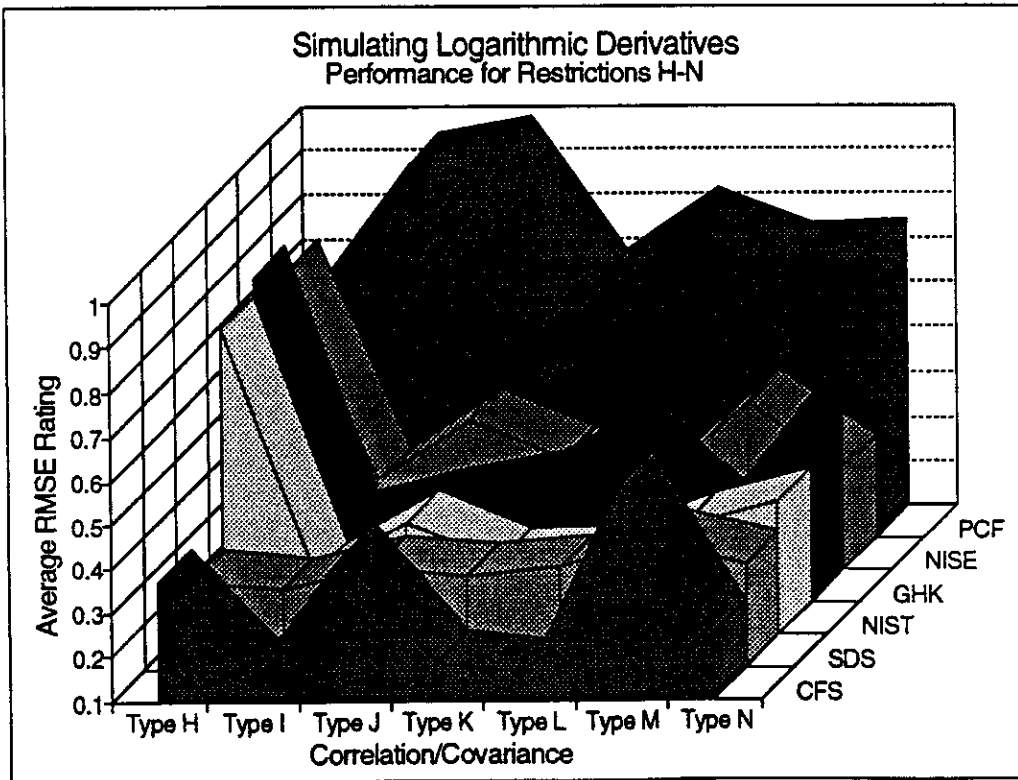
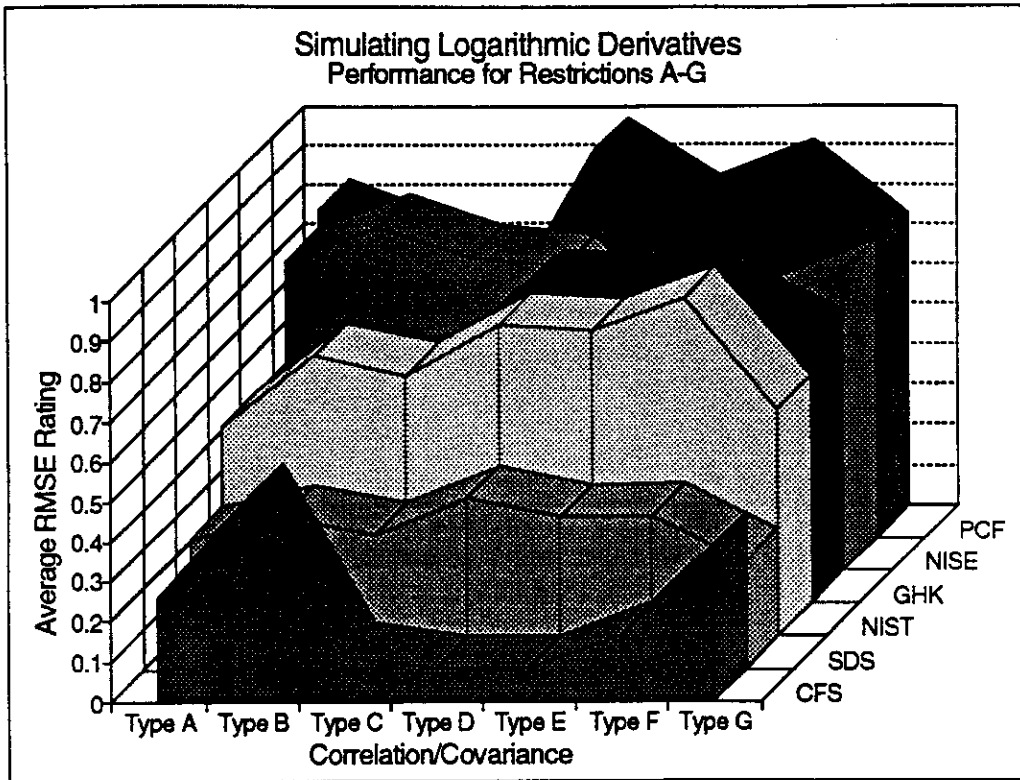


Figure 6 (continued)

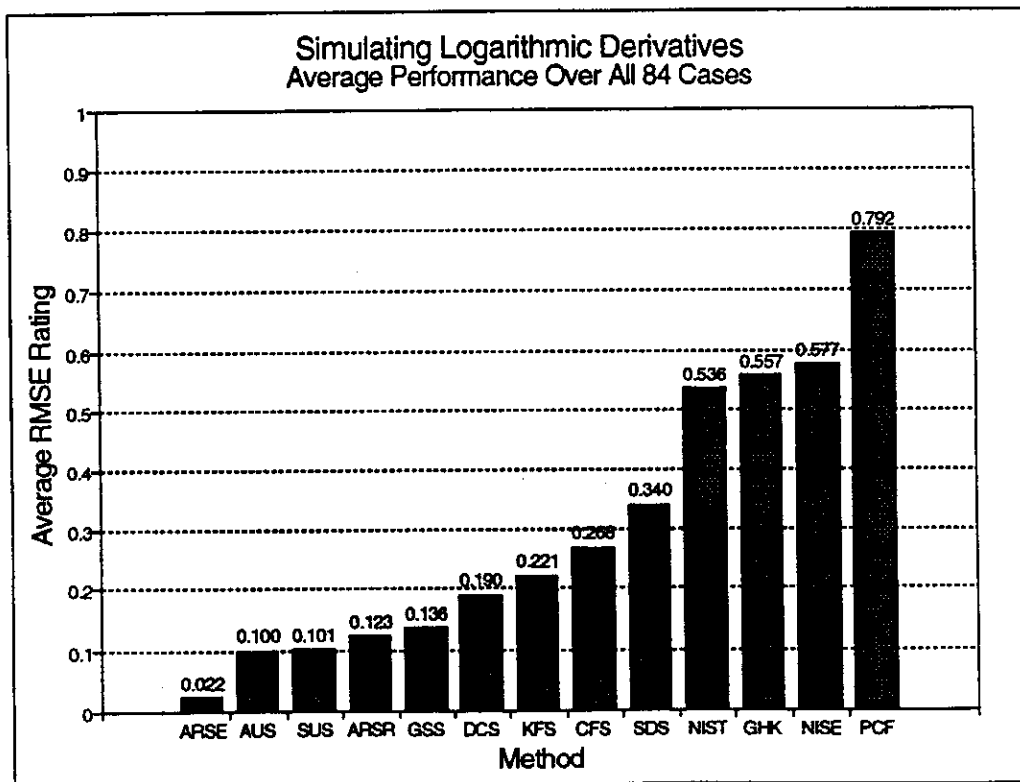
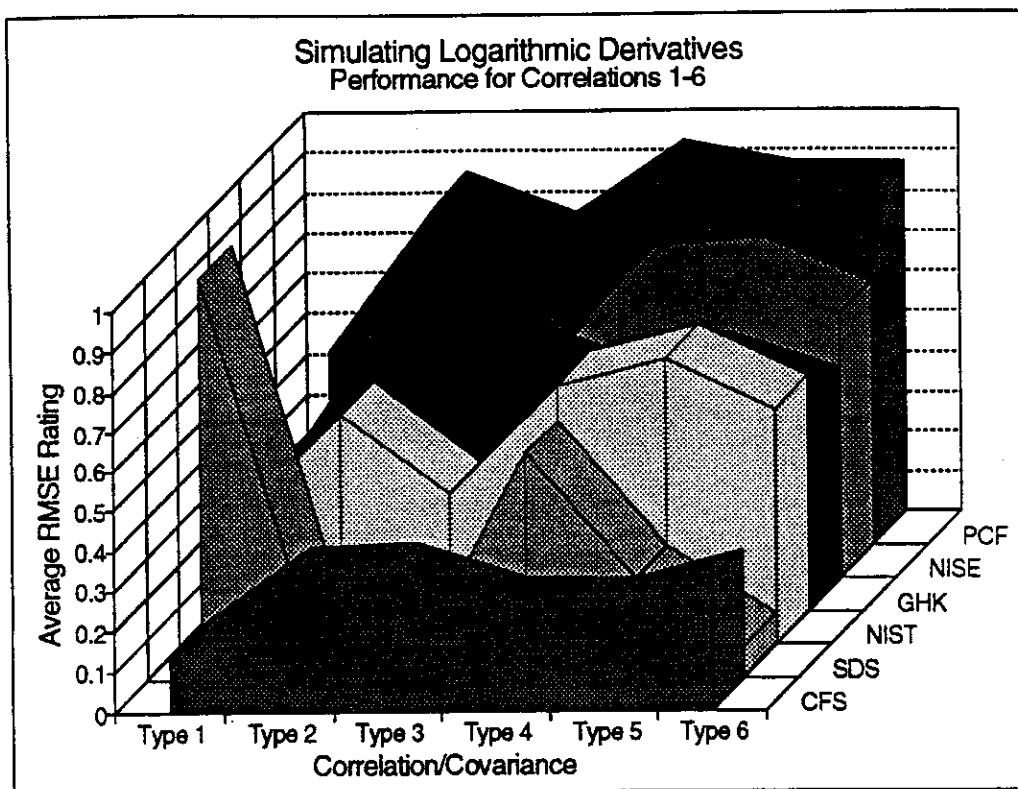


Figure 7
 Changing Number of Simulations: x1, x2, x3, x4, x5

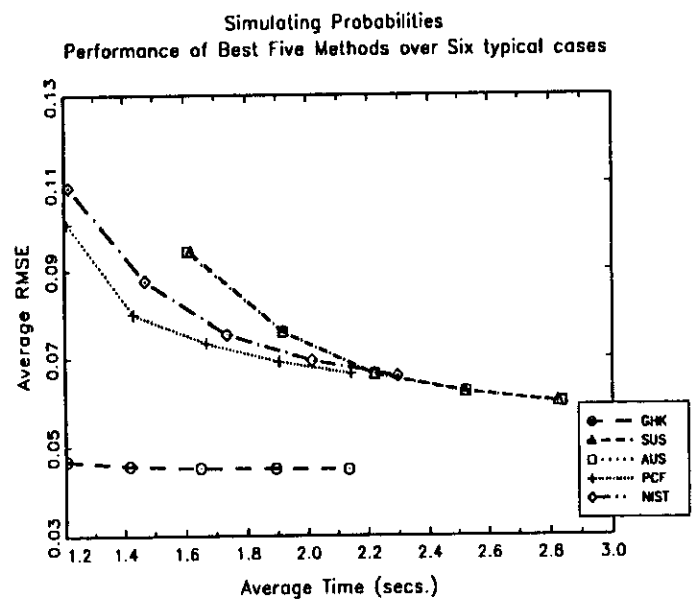
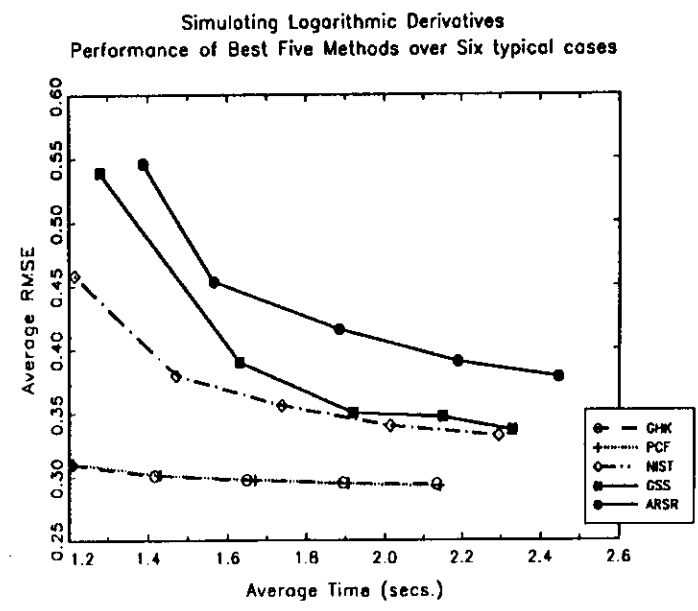
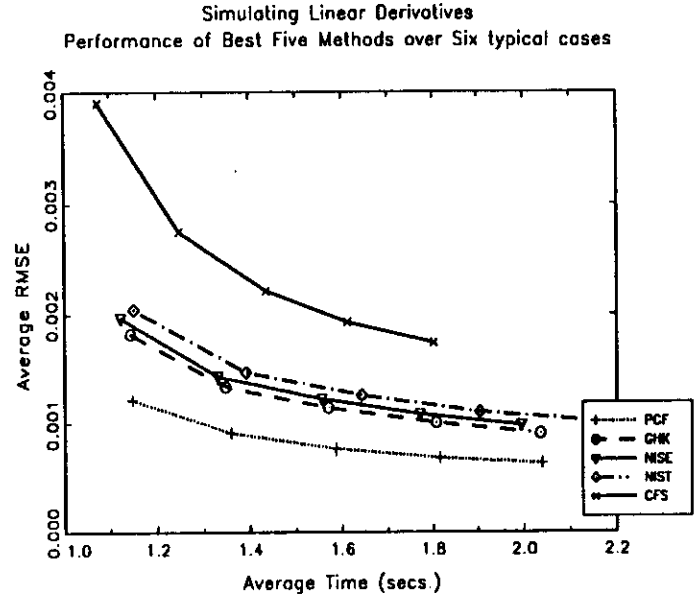


Figure 8
 Increasing Dimension, RE1 Errors: $M = \{2, 4, 8, 16\}$

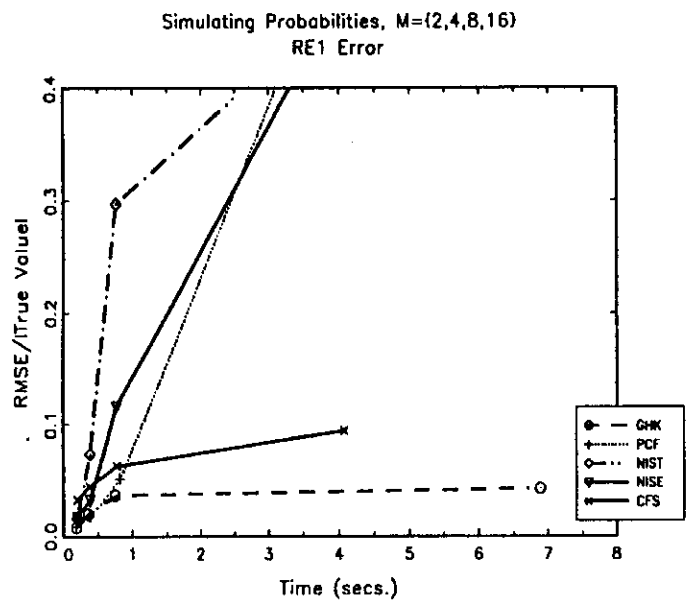
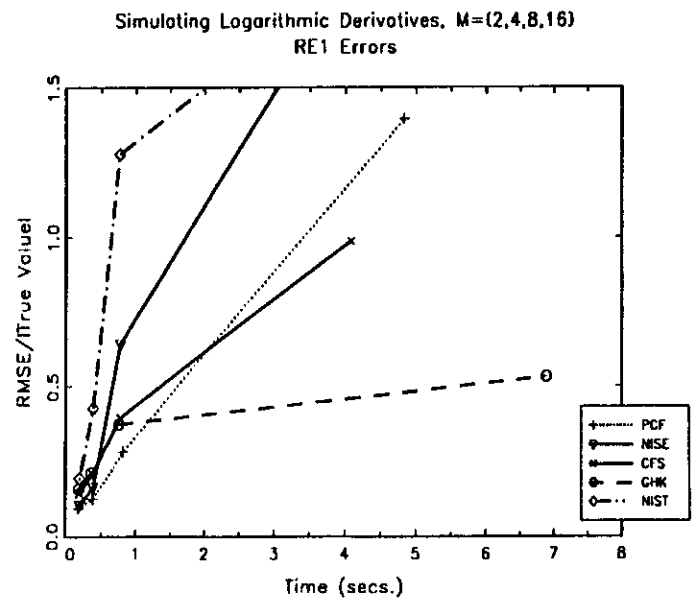
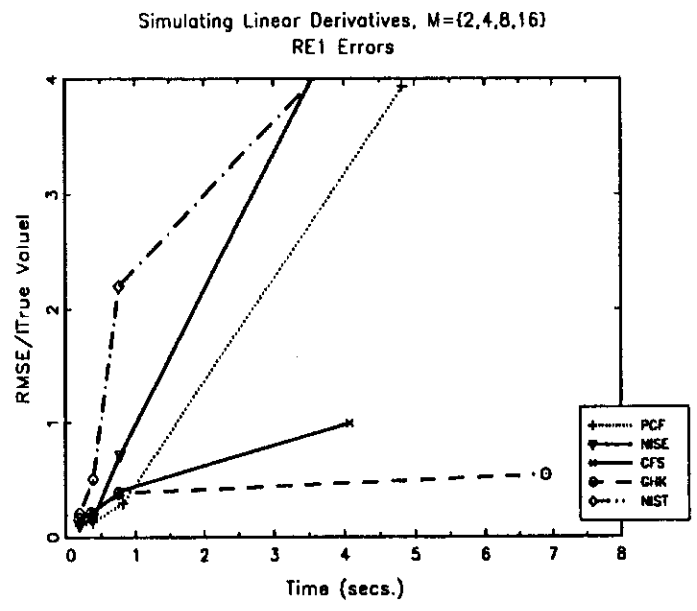


Figure 9
 Increasing Dimension, RE1+AR1 Errors: $M = \{2, 4, 8, 16\}$

