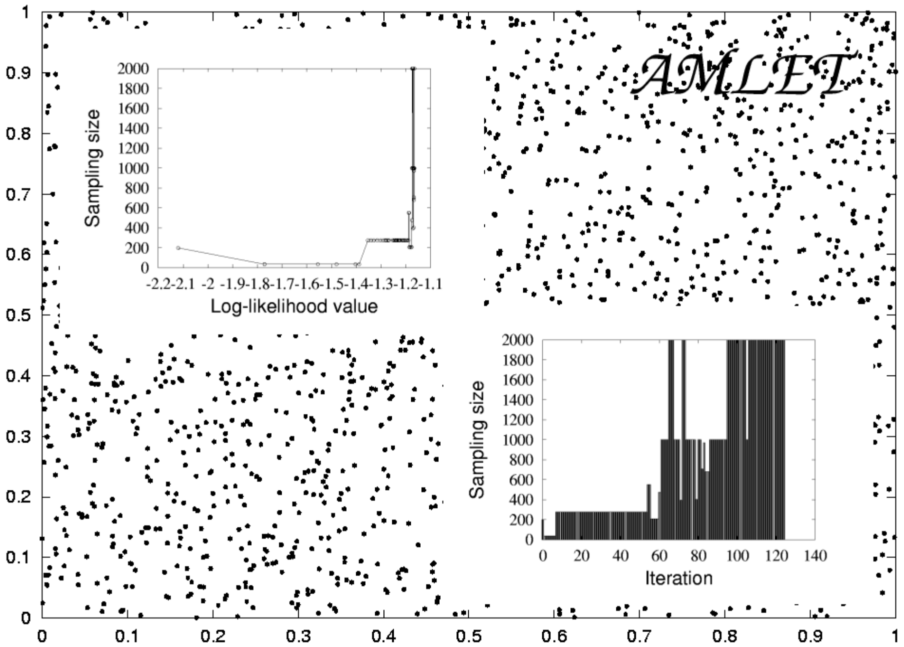


Numerical Experiments with AMLET, a New Monte-Carlo Algorithm for Estimating Mixed Logit Models

Fabian Bastin, University of Namur
Cinzia Cirillo, University of Namur
Philippe L. Toint, University of Namur

Conference paper
Session XXX

Moving through nets:
The physical and social dimensions of travel
10th International Conference on Travel Behaviour Research
Lucerne, 10-15. August 2003

Numerical Experiments with AMLET, a New Monte-Carlo Algorithm for Estimating Mixed Logit Models

Fabian Bastin[1], Cinzia Cirillo and Philippe L. Toint
Department of Mathematics
University of Namur
Namur, Belgium

Phone: 081-72 49 23
Fax: 081-72 49 14
eMail: fabian.bastin@fundp.ac.be

Abstract

Researchers and analysts are increasingly using mixed logit models for estimating responses to forecast demand and to determine the factors that affect individual choices. These models are interesting in that they allow for taste variations between individuals and they do not exhibit the independence from irrelevant alternatives property. However the numerical cost associated to their evaluation can be prohibitive, the inherent probability choices being represented by multidimensional integrals. This cost remains high even if Monte-Carlo techniques are used to estimate those integrals. This paper describes a new algorithm that uses Monte-Carlo approximations in the context of modern trust-region techniques, but also exploits new results on the convergence of accuracy and bias estimators to considerably increase its numerical efficiency. Numerical experiments are presented for both simulated and real data. They indicate that the new algorithm is very competitive and compares favourably with existing tools, including quasi Monte-Carlo techniques based on Halton sequences.

Keywords

Mixed logit, Monte Carlo, stochastic programming, Trust-region, International Conference on Travel Behaviour Research, IATBR

Preferred citation

Bastin, Fabian, Cinzia Cirillo and Philippe L. Toint (2003) Numerical Experiments with AMLET, a New Monte-Carlo Algorithm for Estimating Mixed Logit Models, paper presented at the 10 th International Conference on Travel Behaviour Research, Lucerne, August 2003.

[1]Research Fellow of the Belgian National Fund for Scientific Research

1. Introduction

Discrete choice modelling is a powerful technique for describing how individuals perform a selection amongst a finite set of alternatives. In particular, the multinomial logit and its extensions (see Bhat and Kopelman (1999) for a review in the context of travel demand) are widely used, but the more powerful mixed logit modelling is gaining acceptance among practioners and researchers. However, the numerical cost associated with the evaluation of mixed logit models is significant. In particular the inherent choice probabilities are represented by multidimensional integrals which can only be approximated, in real applications, by Monte-Carlo techniques. Unfortunately, the evaluation costs can still be prohibitive, even when using this technique, due to the required sampling sizes (Hensher and Green, 2003). As a consequence, current research has turned to the cheaper quasi Monte-Carlo approaches: Bhat (2001) and Train (1999) for instance advocate using Halton sequences (Halton, 1960) for mixed logit models and find they perform much better than pure random draws in simulation estimation, while other approaches (Sándor and Train, 2002) attempt to find better quasi Monte-Carlo methods.

This trend is not without drawbacks. For instance, Bhat (2001) recently pointed out that the coverage of the integration domain by Halton sequences rapidly deteriorates for high integration dimensions and consequently proposed a heuristic based on the use of scrambled Halton sequences. He also randomized these sequences in order to allow the computation of the simulation variance of the model parameters. By contrast, the dimensionality problem is irrelevant in pure Monte-Carlo methods, which also benefit from a credible theory for the convergence of the calibration process, as well as of stronger statistical foundations (see for instance Fishman (1996) for a general review, Rubinstein and Shapiro (1993), Shapiro (2000) for application to stochastic programming). In particular, statistical inference on the optimal value is possible, while the quality of the results can only be estimated in practice, for quasi Monte-Carlo procedures, by repeating the calibration process on randomized samples and by varying the number of random draws.

These difficulties led us to reinvestigate pure Monte-Carlo methods for mixed logit estimation, with a special emphasis on numerical efficiency. We propose here a new algorithm for stochastic programming using Monte-Carlo methods, that is based on the trust-region technique. Trust-region methods are well-known in nonlinear nonconvex optimization, and have been proved to be reliable and efficient for both constrained and unconstrained problems. Moreover, the associated theoretical corpus is extensive (Conn *et al.*, 2000). Our efficiency objective led us to adapt the traditional deterministic trust-region algorithm to handle stochasticity and, more importantly, to allow an adaptive variation of the number of necessary random draws. This technique results in an algorithm that is numerically competitive with existing tools for mixed logit models, while giving more information to the practitioner.

Our exposition is organized as follows. We briefly review the mixed logit problem and some of its properties in Section 2. We then introduce our new algorithm in Section 3. Section 4 presents our numerical experimentation and discusses its results. Some conclusions and perspectives are finally outlined in Section 5.

2. The mixed logit problem

Discrete choice models provide a description of how individuals perform a selection amongst a finite set of alternatives. Let I be the population size and $\mathcal{A}(i)$ the set of available alternatives for individual i, $i = 1, \dots, I$. For each individual i, each alternative A_j, $j = 1, \dots, |\mathcal{A}(i)|$, has an associated utility, depending on the individual characteristics and the relative attractiveness of the alternative, which is assumed to have the form

$$U_{ij} = V_{ij} + \epsilon_{ij}, \tag{1}$$

where $V_{ij} = V_{ij}(\beta_j, x_{ij})$ is a function of a vector of model parameters β_j and of x_{ij}, the observed attributes of alternative A_j, while $\boldsymbol{\epsilon}_{ij}$ is a random term reflecting the unobserved part of the utility. Without loss of generality, β_j may be assumed constant across alternatives (i.e. $\beta_j = \beta$ for all j) but the assumption that this vector is identical for all individuals i is crucial and undesirably strong.

The theory then assumes that individual i selects the alternative that maximizes his/her utility. The particular form of the choice probability thus depends on the random terms $\boldsymbol{\epsilon}_{ij}$ in (1). If we assume that they are independently Gumbel distributed with mean 0 and scale factor 1.0, the probability that the individual i chooses alternative j can be expressed with the logit formula

$$L_{ij}(\beta) = \frac{e^{V_{ij}(\beta)}}{\sum_{l=1}^{|\mathcal{A}(i)|} e^{V_{il}(\beta)}}, \tag{2}$$

where we have simplified our notation by dropping the explicit mention of the dependence of L_{ij} and V_{ij} on the known observations x_{ij}. Formula (2) characterizes the classical multinomial logit model.

Mixed logit models relax the assumption that the parameters β are the same for all individuals, by assuming instead that individual parameters vectors $\beta(i)$, $i = 1, \ldots, I$, are realizations of a random vector $\boldsymbol{\beta}$. We assume then that $\boldsymbol{\beta}$ is itself derived from a random vector $\boldsymbol{\gamma}$ and a parameters vector θ, which we express $\boldsymbol{\beta} = \beta(\boldsymbol{\gamma}, \theta)$. For example, if $\boldsymbol{\beta}$ is a K-dimensional normally distributed random vector, we may choose $\boldsymbol{\gamma} = (\boldsymbol{\gamma}_1, \boldsymbol{\gamma}_2, \ldots, \boldsymbol{\gamma}_K)$, with $\gamma_k \sim N(0,1)$[2], and let θ specify the means and standard deviations of the components of $\boldsymbol{\beta}$. The probability choice is then given by

$$P_{ij}(\theta) = E_P\left[L_{ij}(\boldsymbol{\gamma}, \theta)\right] = \int L_{ij}(\gamma, \theta) P(d\gamma) = \int L_{ij}(\gamma, \theta) f(\gamma) d\gamma, \tag{3}$$

where P is the probability measure associated with γ and $f(\cdot)$ is its distribution function.

The vector of parameters θ is then estimated by maximizing the log-likelihood function, i.e. by solving the program

$$\max_\theta LL(\theta) = \max_\theta \frac{1}{I} \sum_{i=1}^{I} \ln P_{ij_i}(\theta), \tag{4}$$

where j_i is the alternative choice made by the individual i. This involves the computation of $P_{ij_i}(\theta)$ for each individual i, $i = 1, \ldots, I$, which is impractical since it requires the evaluation of one multidimensional integral per individual. The value of $P_{ij_i}(\theta)$ is therefore replaced by a Monte-Carlo estimate obtained by sampling over γ, and given by

$$SP^R_{ij_i}(\theta) = \frac{1}{R} \sum_{r=1}^{R} L_{ij_i}(\gamma_r, \theta),$$

where R is the number of random draws γ_r, taken from the distribution function of $\boldsymbol{\gamma}$. As a result, θ is now computed as the solution of the simulated log-likelihood problem

$$\max_\theta SLL^R(\theta) = \max_\theta \frac{1}{I} \sum_{i=1}^{I} \ln SP^R_{in_i}(\theta). \tag{5}$$

We will denote by θ^*_R a solution of this last approximate problem (often called the Sample Average Approximation, or SAA), while θ^* denotes a solution of the true problem (4).

[2] $N(\mu, \sigma)$ stands for the normal distribution with mean μ and standard deviation σ.

2.1 Convergence and useful estimators

Bastin *et al.* (2003) have shown that the mixed logit problem can be viewed as a generalization of the classical stochastic programming problem, which in turn implies that the estimators derived from the SAA problem converge almost surely towards the true maximum likelihood estimators as the sampling size R tends to infinity. For a fixed population size (as is the case in most real applications), they assume that

A.0 the random draws are independently and identically distributed, both for each individual and across them,

A.1 the solutions θ_R^* of the SAA problems (5) remain in some convex compact set S for all R sufficiently large,

A.2 the utilities $V_{ij}(\gamma, \cdot)$, $i = 1, \ldots, I$, $j = 1, \ldots, N$, are continuously differentiable for almost every γ,

A.3 for $t = 1, \ldots, m$, $\frac{\partial}{\partial \theta_t} V_{ij}(\gamma_r, \theta, x_{ij_i})$, $j = 1, \ldots, N$, is dominated by a P-integrable function.

They then deduce that θ_R^* converge, for R tending to infinity, to a vector θ^* which is first-order critical point of for the true problem (4) almost surely.

It is furthermore possible to estimate the error made by using the SAA problem (5) instead of the true problem (4). If we consider a fixed population size and take an independently and identically distributed sample for each individual, it is possible to show that

$$LL(\theta) - SLL^R(\theta) \Rightarrow N\left(0, \frac{1}{I}\sqrt{\sum_{i=1}^{I} \frac{\sigma_{ij_i}^2(\theta)}{R\left(P_{ij_i}(\theta)\right)^2}}\right) \tag{6}$$

as R tends to infinity, where $\Rightarrow$ represents convergence in distribution and where σ_{ij_i} is the standard deviation of $P_{ij_i}(\theta)$. Therefore, $SLL^R(\theta)$ is an asymptotically unbiased estimator of $LL(\theta)$, and the asymptotic value of the confidence interval radius is given by

$$\epsilon_\delta = \alpha_\delta \frac{1}{I}\sqrt{\sum_{i=1}^{I} \frac{\sigma_{ij_i}^2(\theta)}{R\left(P_{ij_i}(\theta)\right)^2}}, \tag{7}$$

where α_δ is the quantile of a $N(0,1)$, associated with some level of signification δ. In practice, one typically chooses $\alpha_{0.9} \approx 1.64$ or $\alpha_{0.95} \approx 1.96$ and evaluates ϵ_δ by replacing $\sigma_{ij_i}(\theta)$ and $P_{ij_i}(\theta)$ by their SAA estimators $\sigma_{ij_i}^R(\theta)$ and $P_{ij_i}^R(\theta)$.

Finally, the simulation bias for finite R can be approximated by the quantity

$$E[SLL^R(\theta)] - LL(\theta) = -\frac{I\epsilon_\delta^2}{2\alpha_\delta^2}.$$

Details of these derivations can be again found in Bastin *et al.* (2003).

3. Optimization algorithm

Solving the SAA problem (5) is very expensive even on modern computers, as pointed for instance by Hensher and Green (2003), since I, the number of multidimensional integrals in the expression of the objective function, can be large (usually in the thousands). Quasi Monte-Carlo methods are appealing to deal with this difficulty since they usually require less random draws to compute these integrals than Monte-Carlo methods for the same accuracy. Their superiority in high dimensions is however less clear and, at variance with Monte-Carlo techniques, they do not allow statistical inference, which prevents computing the accuracy of the objective approximation. We therefore choose to return to Monte-Carlo methods and design an efficient algorithm that exploits statistical inference to limit the number of draws needed in the early iterations, away from the solution. This algorithm is of the trust-region type (see Conn *et al.* (2000) for details and an extensive bibliography on these methods).

3.1 A trust-region algorithm with dynamic accuracy

The main idea of a trust-region algorithm is to calculate a trial point by minimizing a model of the objective function inside a trust region at each iteration. At iteration k, this region is defined as

$$\mathcal{B}_k = \{\theta \in \mathbb{R}^m \,|\, \|\theta - \theta_k\| \leq \Delta_k\},$$

where Δ_k is called the trust-region radius. The predicted and actual reductions in objective function values are then compared. If the agreement is sufficiently good, the trial point becomes the new iterate and the trust-region radius is (possibly) enlarged. If this agreement is poor, the trust region is shrunk in order to improve the quality of the model. In addition, if the model approximates the objective function well compared to the accuracy of the objective function itself (which is dependent on the Monte-Carlo sampling size), we surmise that we could work with a less precise approximation and therefore reduce the sampling size. On the other hand, if the model adequation is poor compared to the precision of the objective function, we increase the sampling size in an attempt to correct this deficiency.

A formal description of our algorithm follows.

Step 0. Initialization. An initial point θ_0 and an initial trust-region radius Δ_0 are given. Define a maximum sampling size $R_{\max}$ and set $R = \max(36, \lceil 0.1\, R_{\max} \rceil)$, where $\lceil x \rceil$ denotes the smallest integral value not less than x. Compute $-SLL^R(\theta_0)$ and set $k = 0$.

Step 1. Model definition. Define in $\mathcal{B}_k$ the quadratic model

$$m_k(\theta_k + s) = SLL^R(\theta_k) + \langle \nabla_\theta SLL^R(\theta_k), s \rangle + \frac{1}{2}\langle s, H_k s \rangle,$$

where H_k is a symmetric approximation to $\nabla_{\theta\theta} f(\theta_k)$. We use the symmetric rank-one (SR1) method to obtain such an approximation, as described by Nocedal and Wright (1999), page 204. Despite being numerically cheap, the SR1 method is known to be efficient in the context of trust-region methods (Conn *et al.*, 1991, 1996; Byrd *et al.*, 1996).

Step 2. Step calculation. Compute a step s_k with the Steihaug-Toint method (see for instance Conn *et al.* (2000), Section 7.5.1, or Nocedal and Wright (1999), page 75). Compute a new adequate sampling size R_k.

Step 3. Acceptance of the trial point. Compute $SLL^{R_k}(\theta_k + s_k)$ and define

$$\rho_k = \frac{SSL^{R_k}(\theta_k + s_k) - SLL^R(\theta_k)}{m_k(\theta_k + s_k) - m_k(\theta_k)}.$$

If $\rho_k < \eta_1$, compare R_k and R. If $R_k > R$, compute $SLL^{R_k}(\theta_k)$, set $R = R_k$ and redefine

$$\rho_k = \frac{SSL^{R_k}(\theta_k + s_k) - SLL^{R_k}(\theta_k)}{m_k(\theta_k + s_k) - m_k(\theta_k)}.$$

If $R_k < R$, compute $SLL^R(\theta_k + s_k)$, set $R_k = R$ and redefine

$$\rho_k = \frac{SSL^R(\theta_k + s_k) - SLL^R(\theta_k)}{m_k(\theta_k + s_k) - m_k(\theta_k)}.$$

If $\rho_k \geq 0.01$, then define $\theta_{k+1} = \theta_k + s_k$ and set $R = R_k$. Otherwise define $\theta_{k+1} = \theta_k$.

Step 4. Trust-region radius update. Set

$$\Delta_{k+1} = \begin{cases} \min\left(10^{20}, \max(2s_k, \Delta_k)\right) & \text{if } \rho_k \geq 0.75, \\ \gamma_1 \Delta_k & \text{if } \rho_k \in [0.01, 0.75), \\ \gamma_2 \Delta_k & \text{if } \rho_k < 0.01. \end{cases}$$

Increment k by 1 and go to Step 1.

In practice we set γ_1 and γ_2 to 0.5. The possible modification of the sampling sizes in Step 3 comes from the fact that if R_k is not equal to R, the computation of $SLL^{R_k}(\theta_k + s_k) - SLL^R(\theta_k)$ can be biased by the differences of precision. In particular, $SLL^{R_k}(\theta)$ can be superior to $SLL^R(\theta_k)$ for all θ in a neighborhood of θ_k. Our technique ensures comparable accuracy for both functions and therefore excludes rejecting iterates due to differing levels of precision.

3.2 Variable sampling size strategy

We now propose a variable sampling size strategy for choosing R_k at Step 2 of the trust-region algorithm, which is described in the following procedure, where

$$\Delta m_k := m_k(\theta_k + s_k) - m_k(\theta_k)$$

denotes the model improvement.

Step 0. Estimate the size needed to obtain a precision on the log-likelihood function equal to the model improvement:

$$R_{\text{suggested}} = \left\lceil \frac{\alpha_\delta^2}{(I\Delta m_k)^2} \sum_{i=1}^{I} \frac{\sigma_{n_i}^2(\theta)}{(P_{in_i}(\theta))^2} \right\rceil.$$

Compute the ratio between the model improvement and the estimated accuracy:

$$\rho_R = \frac{m_k(\theta_k) - m_k(\theta_k + s_k)}{\epsilon_\delta}.$$

If $\rho_R \geq 1.0$ go to Step 1. Otherwise go to Step 2.

Step 1. Since the improvement is larger than the precision, a smaller sampling size is considered:

$$R_k = \max\left(R_{\min}, \min\left(R_{\text{suggested}}, \lceil 0.5\, R_{\max} \rceil\right)\right).$$

Return R_k.

Step 2. Set

$$R_k = \begin{cases} \max\left(R_{\min}, \lceil 0.5\, R_{\max} \rceil\right) & \text{if } \rho_R \geq 0.2, \\ R_{\max} & \text{otherwise.} \end{cases}$$

A sufficient improvement during several consecutive iterations can lead to an overall significant improvement, even if for one iteration it is less than the achieved precision: if the total improvement obtained for several successive iterations is more important than the improvement obtained with a bigger sampling size for one iteration, it can be cheaper to work longer with a smaller sampling size. Therefore we continue to use smaller sampling sizes as long as the model improvement is superior to some threshold, as described in Step 2.

If this variable sampling size technique is used in the trust-region method described in the previous paragraph, we refer to the resulting algorithm as the BTRDA (for Basic Trust-Region with Dynamic Accuracy), while we call algorithm BTR (for Basic Trust-Region algorithm) the same algorithm where the maximum sampling size is used at every iteration, and no estimation of the log-likelihood error is computed or used.

3.3 Stopping tests

Classical stopping criteria usually lead to final iterations that produce objective reductions that are, for too small sampling rates, insignificant compared to the approximation's accuracy. We therefore relax them by stopping the iterative process only if either the maximum sampling size $R_{\max}$ is used or the estimated log-likelihood accuracy is sufficiently small, and if

$$\|\nabla_\theta SLL^R(\theta)\| \leq \max(0.2\epsilon, tol),$$

where ϵ is the estimated log-likelihood accuracy. We also stop the algorithm if a (user preset) maximum number of iterations has been reached without convergence, or if the norm of computed step falls under a user-defined significance threshold (we used 10^{-6}).

4. Numerical investigations

We have implemented our algorithm in C, as part of a new package AMLET (for Another Mixed Logit Estimation Tool). AMLET was tested on a Pentium IV 2 Ghz. We compared our results with those given by Gauss 5.1 and the MaxLik module (Schoenberg, 2001). We have then used Halton sequences and the code written by Train (1999). Due to limitations of our license, we had to use for this test the Pentium III 600 Mhz on which it was originally installed. In order to adequately compare computation times, we ran AMLET on this older computer and found that it is, on this type of applications, approximately 2.7 times slower than the more recent machine. As not all problems could be tested with AMLET on the Pentium III due to memory limitations, we report Gauss times in our comparison after dividing them by this relative speed factor: we write these corrected timings in parenthesis next to the real timing in our tables. We use a signification level δ equal to 0.9 in the estimation of standard deviations and biases.

Our numerical experimentations consider mixed logit models with fixed and normally distributed parameters and involve both simulated and real data. Unless otherwise stated, the starting point for optimization procedures is defined by setting all components to 0.1. For convenience, we will use the abbreviations Halt. and MC for, respectively, Halton and Monte-Carlo.

4.1 Simulations

In our simulations, the attribute values are drawn from a standard univariate normal distribution $N(0, 1)$. The coefficients of each independent variable is also drawn from an univariate normal distribution $N(0.5, 1)$.

The error term is generated from an extreme value (Gumbel) distribution, ensuring that the conditional choice probability follows the logit formula (2). The individual choice is then identified for each observation on the basis of the alternative with the highest utility.

Two simulated experiments varying the number of observations and of alternatives provide the framework of our evaluation. Both test the methodology on four separate integration dimensions (2, 5, 10 and 15). For each of the simulated cases, we then estimated the model 10 times. We finally computed the average optimization time, accuracy (7), simulation bias (2.1) and Euclidian norm of the standard deviations of the found parameters (RMSE) for the ten simulations.

We first vary the number of observations over 2000, 5000, 7500 and 10000, while the number of choice alternatives is kept fixed to 5. We limit the sample size to 1000, mainly due to memory limitations.

The results then show (Figure 1) that the optimization time increases with the number of observations, at an approximately linear rate, while remaining manageable. The estimated error (Figure 2) decreases with the number of observations, while at the same time the bias remains in the same order. The ratio between the bias and the accuracy therefore increases, which makes the application of the delta method as in (6) more dubious as the number of observations grows for fixed sampling size (as stated from the theoretical point of view by Hajivassiliou and McFadden (1998)). The number of dimensions also produces growth in absolute value of the estimated accuracy and bias.

Figure 1: Evolution of time and RMSE with the number of observations

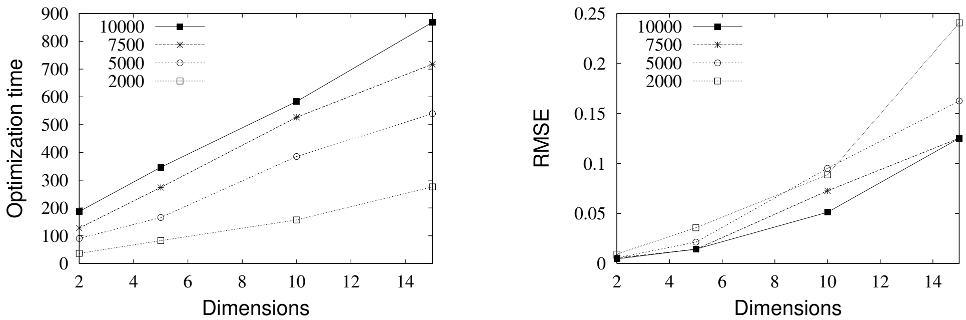
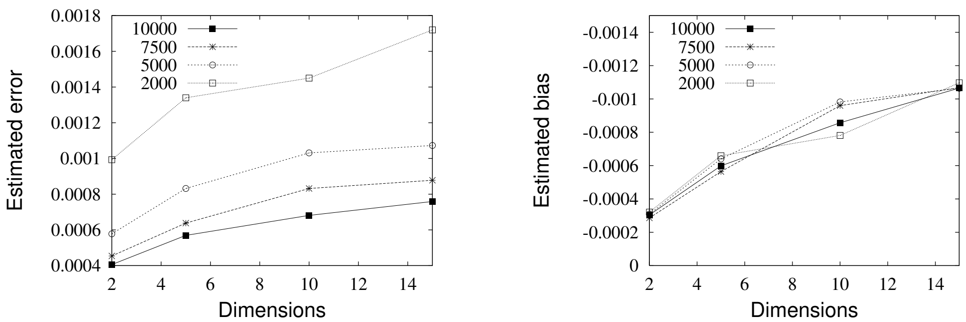


Figure 2: Evolution of estimated error and bias with the number of observations



We next vary the number of alternatives in the choice model amongst 2, 3, 5, and 10 alternatives (one of which is the null alternative); the number of simulated individuals remaining fixed to 5000, and the maximum sampling size is set to 2000.

From Figure 3, we see that the optimization time decreases from 2 to 5 alternatives and then increases from 5 to 10 alternatives. Note also that the RMSE increases with the number of parameters, but the slope is decreased when the number of alternatives grows. A possible interpretation of these phenomena is that having more alternatives makes the discrimination of individual choices easier, leading to a sharper maximum of the simulated likelihood function.

The estimated error grows with the number of alternatives (Figure 4), as well as the bias of simulation, while the RMSE is smaller when the number of alternatives increases. This suggests that the stability of the objective function is far from providing a complete view of the quality of the derived estimations, as already presumed by Bhat (forthcoming). An important topic of research therefore remains to improve error estimates on the parameters when using Monte-Carlo approximations.

Figure 3: Evolution of optimization time and RMSE with the number of alternatives

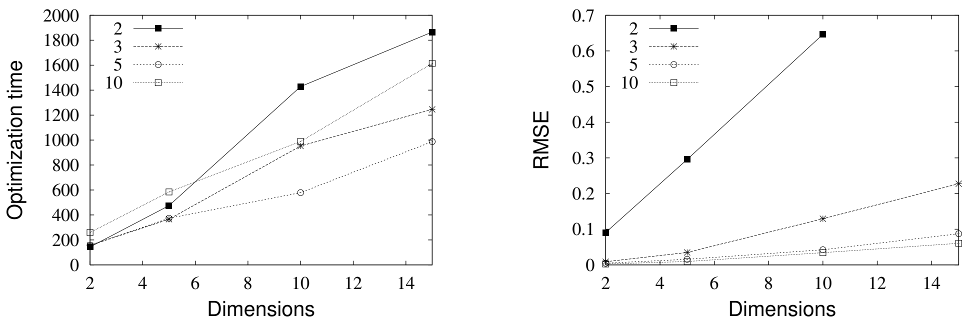
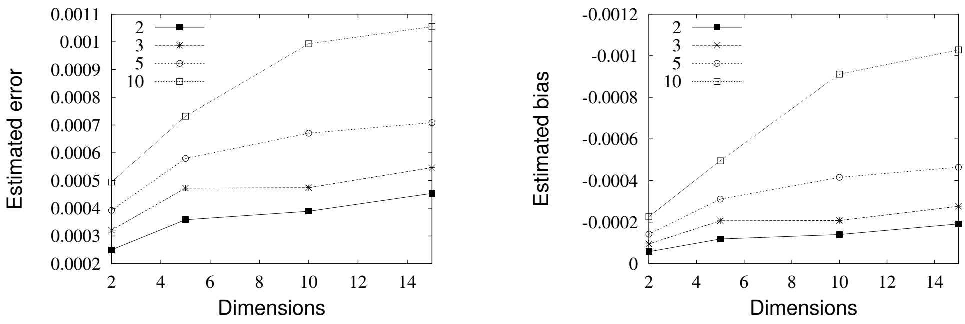


Figure 4: Evolution of estimated error and bias with the number of alternatives



4.2 Comparison of optimization algorithms

Solving (5) requires the use of numerical optimization procedures, amongst which one of the most popular algorithm for calibrating discrete choices models is the BFGS linesearch method (see Train (2003), pages 225–226). It is therefore interesting to evaluate in this context the relevance of trust-region methods, a contending methodology. For comparison purposes, we therefore coded a BFGS algorithm, as described by Nocedal and Wright (1999), using the efficient linesearch technique by Moré and Thuente (1994) and associated code. We then simulate a population of 5000 individuals, facing five alternatives associated with linear utilities with five normal distributed parameters of mean 0.5 and standard deviation 1.0. We used 2000 random draws and compared the BFGS, BTR and BTRDA for solving (5). The procedure was repeated 10 times. Table 1 summarizes the optimal values found for the approximate objective and the corresponding computation times.

Table 1: Comparison of optimization algorithms

Set	BFGS		BTR		BTRDA	
	Likelihood	Time	Likelihood	Time	Likelihood	Time
1	-1.40529	1421	-1.40529	1103	-1.40529	722
2	-1.40532	1455	-1.40532	997	-1.40532	618
3	-1.40519	1691	-1.40519	1090	-1.40519	730
4	-1.40409	1736	-1.40409	997	-1.40409	564
5	-1.40525	1702	-1.40525	1017	-1.40525	531
6	-1.40499	1707	-1.40499	1089	-1.40499	587
7	-1.40437	1475	-1.40437	1045	-1.40437	560
8	-1.40530	1706	-1.40530	998	-1.40530	781
9	-1.40532	1701	-1.40532	1028	-1.40532	724
10	-1.40441	1729	-1.40441	1044	-1.40441	546
Mean	-1.40495	1632	-1.40495	1040	-1.40495	636

While each method gives the same optimal value, BTR requires in average 63.7% of the time used by the BFGS algorithm. The BTRDA algorithm however requires on average 61.2% of the time needed by BTR, and 39.0% of the BFGS time. These results tend therefore to comfort our choice of a trust-region approach combined with the variable sampling size strategy.

4.3 Comparison with Halton sequences

A major drawback of Monte-Carlo methods is their slow convergence rate. Quasi Monte-Carlo methods attempt to speed up this convergence by being more directive in the choice of the sampling points used to evaluate the choice probabilities. This led Train (1999) to suggest the use of Halton draws instead of Monte-Carlo random draws. He found that when the number of random parameters is small (typically less or equal to five), approximation of the objective function based on Halton sequences usually succeeds to give the same results as pure Monte-Carlo sampling, with less random draws. The same conclusion can be drawn from Table 2. In this example, we consider a simulated population of size 2000 and five independent normally distributed parameters, of mean 0.5 and standard deviation 1.0, with individuals faced to five alternatives. Results reported for AMLET are averaged on ten simulations. This table also indicates that AMLET is considerably faster than Gauss, even if larger sample sizes are used.

When the number of random parameters grows, results deteriorate, as illustrated in Table 3, where ten

Table 2: Halton and Monte-Carlo samplings for 5 random parameters

Variable	Gauss		AMLET	
	125 Halt.	250 Halt.	1000 MC	2000 MC
P1 mean	0.4288	0.4287	0.427227	0.429323
P1 std. dev.	1.0854	1.0837	1.06264	1.06692
P2 mean	0.4489	0.4230	0.447468	0.421373
P2 std. dev.	1.1620	1.1622	1.17045	1.18612
P3 mean	0.5704	0.5697	0.567395	0.570794
P3 std. dev.	1.1212	1.1044	1.09255	1.109915
P4 mean	0.6165	0.6212	0.612949	0.617822
P4 std. dev.	1.3386	1.3677	1.33102	1. 34101
P5 mean	0.6548	0.6565	0.645011	0.648492
P5 std. dev.	1.2558	1.2360	1.21059	1.22397
Log-likelihood	-1.44770	-1.44784	-1.44835	-1.44813
Bias	NA	NA	-0.000826983	-0.000415893
Accuracy	NA	NA	0.00149579	0.00106076
Time (s)	959 (355)	1649 (611)	73	148

independent normal distributed parameters are considered, for a simulated population of size 2000. We see that the pure Monte-Carlo approach then gives results that are intuitively better than those produced by Halton techniques (AMLET gives means and standard deviations estimates that are larger than those obtained with Gauss and closer to the parameters used for the data simulation). Moreover, AMLET is again faster than Gauss. We also observe that the Halton sequences results deteriorate when the number of draws increases from 125 to 250, while they seem to improve between 1000 and 2000 random draws for AMLET. This reflects the fact that, at variance with Monte-Carlo techniques, Halton sequences may probably be less suitable for high-dimensional problems. Indeed, the sequences associated with successive prime numbers exhibit stronger correlations as the prime numbers grow and the space is no longer covered as uniformly as the dimension increases.

These problems can possibly be addressed by considering scrambled Halton sequences (Bhat, 2001). It is thus our intention to pursue the work reported here by comparing our Monte-Carlo-based results with those obtained with such techniques.

4.4 Tests on real data

We finally validated our algorithm and tested its performance on a real data set, whose source is the six-week diary Mobidrive (Axhausen *et al.*, 2002), collected in the spring and fall 1999 in Karlsruhe and Halle (Germany). The sample includes about 160 households and about 360 individuals. We only use here data from the city of Karlsruhe itself, for which the level of service variables are available. The case study is a mode choice model across five alternatives: car driver, car passenger, public transport, walk and bike. The framework applied considers the daily activity chain; the individual pattern is divided into tours, which are defined as the sequence of trips starting and ending at home or at work, considered both to be at a fixed location. After careful data cleaning, the number of observations available for estimation is 5799. It must be noted that the panel data contains multiple observations per day and per individual, which induces correlation, a phenomenon that we do not attempt to tackle here. For further details on mixed logit on panel data estimated on the same data set, see Cirillo and Axhausen (2002).

Table 3: Halton and Monte-Carlo sampling for 10 random parameters

Variable	Gauss		AMLET	
	125 Halt.	250 Halt.	1000 MC	2000 MC
P1 mean	0.4382	0.4406	0.429475	0.47396
P1 std. dev.	0.5532	0.6336	0.707418	0.70936
P2 mean	0.3335	0.3352	0.353878	0.357928
P2 std. dev.	0.6735	0.6405	0.712054	0.72369
P3 mean	0.4232	0.4217	0.487211	0.493233
P3 std. dev.	0.7030	0.7383	0.754033	0.768334
P4 mean	0.3815	0.3672	0.391026	0.393294
P4 std. dev.	0.8519	0.8112	0.872773	0.878611
P5 mean	0.3902	0.3772	0.400395	0.405856
P5 std. dev.	0.7650	0.7774	0.830782	0.856396
P6 mean	0.3506	0.3516	0.376615	0.381124
P6 std. dev.	0.7404	0.7317	0.835979	0.845602
P7 mean	0.5321	0.5201	0.556178	0.564064
P7 std. dev.	0.8460	0.8146	0.901123	0.909371
P8 mean	0.4313	0.4229	0.448092	0.423366
P8 std. dev.	0.6720	0.5611	0.633405	0.646381
P9 mean	0.3860	0.3823	0.408147	0.413727
P9 std. dev.	0.5333	0.4264	0.555716	0.571289
P10 mean	0.3801	0.3797	0.404174	0.407996
P10 std. dev.	0.6472	0.7226	0.762068	0.776279
Log-likelihood	-1.42105	-1.42283	-1.42274	-1.42248
Bias	NA	NA	-0.000840932	-0.000422711
Accuracy	NA	NA	0.00150823	0.00106938
Time (s)	1312 (486)	2515 (699)	206	372

The estimated model contains 14 variables, four of which are alternative specific constants (car driver is the base) and 3 dimensions. We estimate household location characteristics (urban location and suburban location), individual socio-demographic variables (female and part time, married with children, annual mileage), a tour variable (number of stops), a pattern variable (time budget) and a level of service variable. Time budget is defined as 24 hours minus the time already spent in previous activities out of home and in home, or travelling. The coefficients are assumed to be identical between individuals, except for time, cost and time budget, that are expected to vary significantly and are therefore represented by normally distributed random variables.

We have calibrated the model with Gauss and 125 Halton draws, and AMLET, with 1000 and 2000 Monte-Carlo random draws. Results for AMLET have been averaged over 10 simulations. Both packages were started from the initial point

$$\begin{aligned}\theta_0 = (&-1.0,\ -1.0,\ 0.0,\ -1.0,\ 0.0,\ -1.0,\ 0.0,\ 0.0,\\ &0.0,\ 0.0,\ 0.0,\ -1.0,\ 0.5,\ -1.0,\ 0.5,\ -1.0,\ 0.5).\end{aligned}$$

Results are summarized in Table 4. The car driver is taken as base, and the first four parameters are specifics constants. The alternatives associated with other specifics parameters are written in brackets, next to the parameter name. The following abbreviations have been used: CD for car driver, CP for car passenger, PT for public transport, W for walk and B for bike.

AMLET reports solutions that are similar to those obtained with Gauss, supporting the observation

Table 4: Tests on real data

Variable	Gauss (125 Halt.)	AMLET (1000 MC)	AMLET (2000 MC)
	Coefficient	Coefficient	Coefficient
Car Passenger (CD)	-1.4511	-1.45104	-1.4527
Public Transport (PT)	-0.9355	-0.932594	-0.932458
Walk (W)	0.1081	0.109186	0.108793
Bike (B)	-0.6355	-0.634269	-0.635217
Urban household location (PT)	0.560609	0.557866	0.561515
Suburban household location (W, B)	-0.3451	-0.345403	-0.345113
Full-time worker (PT)	0.2690	0.269265	0.268996
Female and part-time (CP)	0.9133	0.912925	0.913835
Married with children (CD)	0.9716	0.970755	0.971656
Annual mileage (CD)	0.0518	0.0518679	0.0519161
Number of stop (CD)	0.1349	0.135187	0.135817
Time mean	-0.0268	-0.0268882	-0.0269985
Time std. dev.	0.0205	0.0206265	0.0208197
Cost mean	-0.1683	-0.168923	-0.169365
Cost std. dev.	-0.0452	0.0465829	0.0465628
Time budget/100 mean (CD, CP)	-0.1249	-0.124816	-0.125128
Time budget/100 std. dev. (CD, CP)	-0.1136	0.112801	0.113803
Log-likelihood	-1.16489	-1.16479	-1.16470
Bias	Not available	-0.0000911731	-0.000046262
Accuracy	Not available	0.000291612	0.000208614
Time (s)	6585 (2439)	936	1549

that good results can be obtained with a small number of Halton draws when the number of random parameters is limited. However, the optimization time of AMLET is very competitive with that of Gauss, since they are similar for 2000 random draws and 125 Halton draws. AMLET is clearly faster for a sampling size of 1000. Using the final SR1 Hessian approximation in the computation of the t-statistics, we observed that these statistics are usually similar to those obtained with the exact Hessian, except for two cases. However all simulations, less one (with 2000 random draws) report that all parameters except the third are significant at the level 0.95. Note that the evaluation of the exact Hessian is a very expensive task, especially when the number of parameters is high, while the approximate Hessian is directly available. The potential time saving when using the SR1 approximation could therefore be important: we believe that investigating more precisely when the Hessian approximation is sufficient for computing the t-statistics is a valuable direction for further research.

The crucially beneficial effect of the variable sampling size strategy is illustrated in Figure 5, giving the evolution of the sampling size R_k with the iteration index k. The left graph corresponds to a maximum sample size of 1000 while the right graph has been obtained with a maximum of 2000 random draws. Furthermore, Figure 6 shows that the sampling size rises towards its maximum value only when the objective function's value is near to its maximum. The graphs correspond again to 1000 and 2000 random draws, respectively.

Figure 5: Variation of sample sizes with iterations

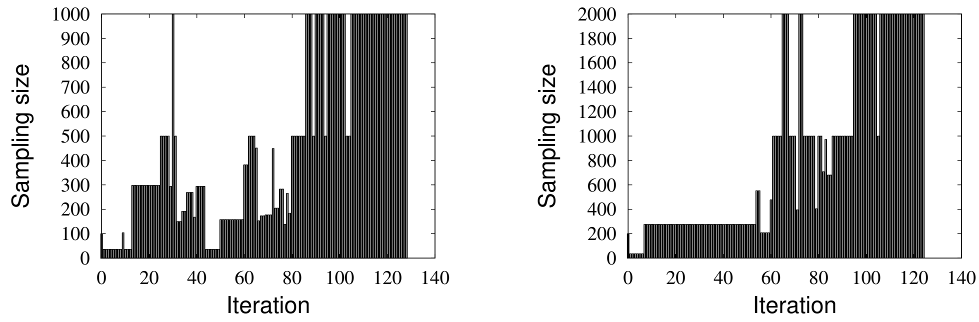
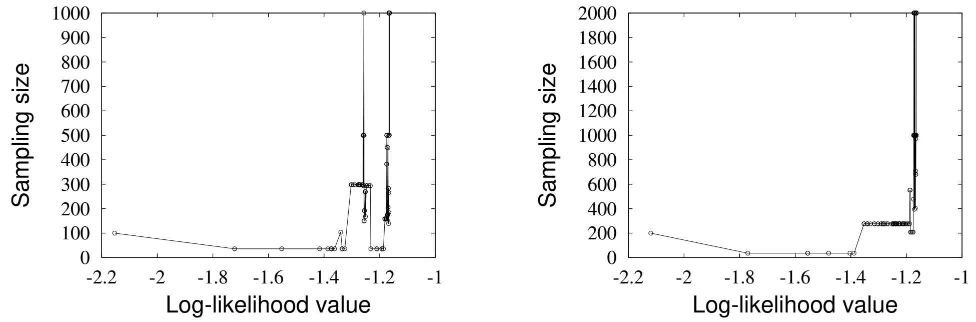


Figure 6: Variation of sample sizes with log-likelihood value



5. Conclusion and perspectives

We have developed a new trust-region algorithm for unconstrained stochastic programming whose efficiency is improved by the use of statistical inference. We have implemented this algorithm in the AMLET package and tested it on simulated and real data for estimating mixed logit models. Numerical experimentation shows that trust-region methods are more efficient than the popular BFGS algorithm and that significant further gains in optimization time can be achieved with the proposed variable sampling size strategy.

We have also shown that our package's timings are competitive with Gauss and Halton sequences while giving information on the quality of the Monte-Carlo approximation and not suffering of non-uniformity in high dimensions. Moreover the proposed techniques can probably be refined to obtain further speed-ups. Simulations suggest however that the accuracy of the log-likelihood does not completely reflect the precision of the estimated parameters. Further research therefore remains important in order to quantify the statistical accuracy of the simulated estimators compared to the true maximum likelihood estimators.

AMLET is still under development. Future features include allowing the user to choose non-normal distributions and to take into account correlations when we have several observations from the same individual.

Acknowledgments

The authors would like to express their gratitude to Michel Bierlaire concerning implementation details for discrete choice models, and to Kay Axhausen for allowing us to use the Mobidrive data set. Thanks are also due to Marcel Remon for his helpful comments on statistical theory, and to the Belgian National Fund for Scientific Research for the grant that made this research possible for the first author.

References

Axhausen, K. W., A. Zimmerman, S. Schönfelder, G. Rindsfüser and T. Haupt (2002) Observing the rythms of daily life: A six week travel diary, *Transportation*, **29** (2) 95–124.

Bastin, F., C. Cirillo and P. L. Toint (2003) Convergence theory for nonconvex stochastic programming with an application to mixed logit, ***Tech. Rep.***, Department of Mathematics, University of Namur, Namur, Belgium. Forthcoming.

Bhat, C. R. (2001) Quasi-random maximum simulated likelihood estimation of the mixed multinomial logit model, *Transportation Research*, **35B** (7) 677–693, Aug. 2001.

Bhat, C. R. (forthcoming) Simulation estimation of mixed discrete choice models using randomized and scrambled Halton sequences, *Transportation Research B*.

Bhat, C. R. and F. S. Kopelman (1999) Activity-based modeling of travel demand., in R. W. Hall (Ed.), *Handbook of Transportation Science*, 35–61, Kluwer Academic Publisher.

Byrd, R. H., H. Fayez Khalfan and R. B. Schnabel (1996) Analysis of a symmetric rank-one trust region method, *SIAM Journal on Optimization*, **6** (4) 1025–1039.

Cirillo, C. and K. W. Axhausen (2002) Mode choice of complex tour, in *Proceedings of the European Transport Conference*, Cambridge, UK.

Conn, A. R., N. I. M. Gould and P. L. Toint (1991) Convergence of quasi-newton matrices generated by the symmetric rank one update, *Mathematical Programming*, **50** (2) 177–196.

Conn, A. R., N. I. M. Gould and P. L. Toint (1996) Numerical experiments with the LANCELOT package (Release A) for large-scale nonlinear optimization, *Mathematical Programming, Series A*, **73** (1) 73–110.

Conn, A. R., N. I. M. Gould and P. L. Toint (2000) *Trust-Region Methods*, Siam, Philadelphia, USA.

Fishman, G. S. (1996) *Monte Carlo: Concepts, Algorithms and Applications*, Springer Verlag, New York, USA.

Hajivassiliou, V. A. and D. L. McFadden (1998) The method of simulated scores for the estimation of LDV models, *Econometrica*, **66** (4) 863–896.

Halton, J. H. (1960) On the efficiency of certain quasi-random sequences of points in evaluating multi-dimensional integrals, *Numerische Mathematik*, **2** 84–90.

Hensher, D. A. and W. H. Green (2003) The mixed logit model: The state of practice, *Transportation*, **30** (2) 133–176.

Moré, J. J. and D. J. Thuente (1994) Line search algorithms with guaranteed sufficient decrease, *ACM Transactions on Mathematical Software*, **20** (3) 286–307.

Nocedal, J. and S. J. Wright (1999) *Numerical Optimization*, Springer, New York, USA.

Rubinstein, R. Y. and A. Shapiro (1993) *Discrete Event Systems*, John Wiley & Sons, Chichester, England.

Sándor, Z. and K. Train (2002) Quasi-random simulation of discrete choice models, *Department of Economics, University of California*.

Schoenberg, R. (2001) Optimization with the quasi-newton method, Aptech Systems, Inc., Maple Valley, WA, USA.

Shapiro, A. (2000) Stochastic programming by monte carlo simulation methods, *SPEPS*.

Train, K. (1999) Halton sequences for mixed logit, Working paper No. E00-278, Department of Economics, University of California, Berkeley.

Train, K. (2003) *Discrete Choice Methods with Simulation*, Cambridge University Press, New York, USA.