

Estimating the parameters of Logit Model using simulated annealing algorithm: case study of mode choice modeling of Isfahan

Sh.Afandizadeh^{1,*}, S.A.H Zahabi² and N.Kalantari³
Received: September 2008 Accepted: January 2010

Abstract: Logit models are one of the most important discrete choice models and they play an important role in describing decision makers' choices among alternatives. In this paper the Multi-Nominal Logit models has been used in mode choice modeling of Isfahan. Despite the availability of different mathematical computer programs there are not so many programs available for estimating discrete choice models. Most of these programs use optimization methods that may fail to optimize these models properly. Even when they do converge, there is no assurance that they have found the global optimum, and it just might be a good approximation of the global minimum. In this research a heuristic optimization algorithm, simulated annealing (S.A), has been tested for estimating the parameters of a Logit model for a mode choice problem that had 17 parameters for the city of Isfahan and has been compared with the same model calculated using GAUSS that uses common and conventional algorithms. Simulated annealing is an algorithm capable of finding the global optimum and also it's less likely to fail on difficult functions because it is a very robust algorithm and by writing the computer program in MATLAB the estimation time has been decreased significantly. In this paper, this problem has been briefly discussed and a new approach based on the simulated annealing algorithm to solve that is discussed and also a new path for using this technique for estimating Nested Logit models is opened for future research by the authors. For showing the advantages of this method over other methods explained above a case study on the mode choice of Isfahan has been done.

Keywords: Simulated annealing, Multi-Nominal Logit, Mode choice, Maximum Likelihood estimation (MLE)

1. Introduction

By far the easiest and most widely used discrete choice model is logit. Its popularity is due to the fact that the formula for the choice probabilities takes a closed form and is readily interpretable[1]. Originally, the logit formula was derived by Luce (1959) from assumptions about the characteristics of choice probabilities, namely the Independence of Irrelevant Alternatives (IIA) property. Numerous computer packages contain

routines for estimation of logit models that use econometric methods, such as nonlinear least squares, the generalized method of moments, the maximum likelihood method, and Newton-Raphson method which rely upon optimization to estimate model parameters. Not only do these methods lay the theoretical foundation for many estimators, they are often used to numerically estimate models. Unfortunately, this may be difficult because these algorithms are sometimes touchy and may fail. For the maximum likelihood method, Cramer (1986, p. 77) lists a number of "unpleasant possibilities" due to the usage of unreliable optimization algorithms based on numerical iteration : the algorithm may not converge in a reasonable number of steps, it may head toward infinitely large parameter values, or even loop through the same point time and again. Also, the algorithm may have difficulty with ridges and plateaus. When faced with such difficulties, the researcher is often reduced to trying different starting values [Cramer, (1986, p.72) and Finch et al. (1989)]. The problem of zig

* Corresponding author. Email: zargari@iust.ac.ir

1 Associate Professor, Department of Civil Engineering, Iran University of Science and Technology, Tehran, Iran.

2 Masters student of Civil Engineering, Department of Civil Engineering, Iran University of Science and Technology, Tehran, Iran.

3 PHD Candidate of Civil Engineering, Department of Civil Engineering, Iran University of Science and Technology, Tehran, Iran.

zaging is a well known problem of most conventional line search optimization algorithms that has been traditionally used for this problem. In sum, there is a poor match between the power of these methods and the numerical algorithms used to implement them. A new and very different optimization algorithm, simulated annealing, potentially improves this match. It explores the function's entire surface and tries to optimize the function while moving both uphill and downhill. Thus, it is largely independent of the starting values, often a critical input in conventional algorithms, because it's basically a probabilistic iteration algorithm. Further, it can escape from local optima and go on to find the global optimum by the uphill and downhill moves. Simulated annealing also makes less stringent assumptions regarding the function than do conventional algorithms (it need not even be continuous). Because of the relaxed assumptions, it can more easily deal with functions that have ridges and plateaus. Finally, it can optimize functions that are not defined for some parameter values [2].

The present study focuses on maximum likelihood method, and uses simulated annealing to maximize the Log likelihood function for estimating the parameters of a Multi-Nominal Logit model (MNL), used for mode choice modeling of the city of Isfahan. Simulated annealing is an algorithm that originates in material science engineering, originally introduced to find the equilibrium configuration of a collection of atoms at a given temperature [3].

The first one to use it as an algorithm to solve optimization problems was Kirkpatrick et al. [4]. This algorithm considers the optimization process as a Markovian chain that pushes the iterations towards a minimum or maximum. At each iteration or better to say each temperature value, the algorithm checks a number of finite different situations that the Markovian chain can be in to find a new point to move to. This procedure should be controlled with great care to assure the convergence towards a global maximum or minimum. But it should be taken into account that there is no need to have in depth knowledge on the Markovian chains. Altogether,

it is a very good and easy to implement algorithm that helps engineers find their way through hard optimization problems.

The next section deals with logit models and specially MNL. Section 3 deals with parameter estimation using MLE. Section 4 gives some brief information about what the SA algorithm is, and the way it can be applied for parameter estimation for the problem at hand. The following section illustrates the approach here presented focusing on a case study for the mode choice modeling of Isfahan to prove the advantages of using S.A. Part 6 concludes the current work followed by a section introducing some research areas to those interested in parameter estimation and opens a new horizon for using this approach in estimating Nested Logit models.

2. Logit Models

The principles of individual choice models are based upon Random Utility Theory. There are different types of models for simulating individual's choices. For example there are: Multi-Nominal Logit models, Nested Logit Models, Cross Nested Logit Models, Mixed Logit Models and Probit Models.

The easiest and most widely used discrete choice model is logit. Its popularity is due to the fact that the formula for the choice probabilities takes a closed form and is readily interpretable. A decision maker, labeled n , faces J alternatives. The utility that the decision maker obtains from alternative j is decomposed into (1) a part labeled V_{nj} that is known by the researcher up to some parameters, and (2) an unknown part ε_{nj} that is treated by the researcher as random: $U_{nj} = V_{nj} + \varepsilon_{nj} \quad \forall j$. The Multi-Nominal logit model is obtained by assuming that each ε_{nj} is independently, identically distributed extreme value. The distribution is also called Gumbel and type I extreme value (and sometimes, mistakenly Weibull). The density for each unobserved component of utility is [1]:

$$f(\varepsilon_{nj}) = e^{-\varepsilon_{nj}} e^{-e^{-\varepsilon_{nj}}} \quad (1)$$

And the cumulative distribution is [1]:

$$F(\varepsilon_{nj}) = e^{-e^{\varepsilon_{nj}}} \quad (2)$$

In this model the probability that person n chooses the alternative i from a subset of alternatives C_n is [1]:

$$P_{n,i} = \frac{e^{V_{ni}}}{\sum_{j \in C_n} e^{V_{nj}}} \quad (3)$$

Representative utility is usually specified to be linear in parameters: $V_{nj} = \beta X_{nj}$, where X_{nj} is a vector of observed variables relating to alternative j . With this specification, the logit probabilities become [1]:

$$P_n = \frac{e^{\beta' X_{ni}}}{\sum_{j \in C_n} e^{\beta' X_{nj}}} \quad (4)$$

Importantly, McFadden (1974) demonstrated that the log-likelihood function with these choice probabilities is globally concave in parameters β , which helps in the numerical maximization procedures, but most conventional optimization algorithms cannot find the optimum value due to the zig zaging of the algorithms and that these algorithms terminate when the improvements in the objective function is not significant, a short coming that most heuristic algorithms have overcome and in this paper we will emphasize on this property and show how it can improve the results.

In aggregate models it is assumed that all variables in the probability function are the same for all the people in the sample population ($X_{ni} = X_i$). By this assumption we have:

$$P_{n,i} = P_i \quad (5)$$

In this study we have used aggregate MNL to model the mode choice of our case study because of the ease of computation and the ability to compare our results with estimations done in the Second Comprehensive Studies of Transportation and Traffic Isfahan [21].

3. Parameter Estimation Using MLE

Estimation theory is a cornerstone in statistical analyses, and it has been introduced several

techniques to estimate parameters, of which MLE (maximum likelihood estimation), graphical procedure [5], moments method [6,7] and weighted least square method are some of the most interesting ones [8].

The MLE method has very desirable properties. Let x_1, x_2, \dots, x_n be a random sample of size n drawn, at random, from a probability density function, $f(x; \vec{\theta})$ of unknown parameters, $\vec{\theta}$. The likelihood function is as follows:

$$L = \prod_{i=1}^n f_{x_i}(x_i; \vec{\theta}) \quad (6)$$

Where $\vec{\theta}$ is a vector of size m representing the unknown parameters, i.e.:

$$\vec{\theta} = (\theta_1, \theta_2, \dots, \theta_m) \quad (7)$$

The aim here is to find a vector that maximizes the so called likelihood function. To maximize L we can use its logarithm, $\ln(L)$, and maximize that instead for a more accurate estimation. Estimates are obtained by solving the following equation:

$$\frac{\partial}{\partial \vec{\theta}} \ln(L) = \vec{0} \quad (8)$$

Where $\vec{0}$ is a null vector (all elements equal to Zero).

The above equation set is solved using different computer programs. Most of these programs use optimization methods that may fail. Even when they do converge, there is no assurance that they have found the global optimum and not a near optimum solution. In this regard we have used a new and very different optimization algorithm, simulated annealing, that potentially improves this match. It explores the function's entire surface and tries to optimize the function while moving both uphill and downhill. Thus, it is largely independent of the starting values, often a critical input in conventional algorithms because it's basically a probabilistic iteration algorithm. Further, it can escape from local optima and go on to find the global optimum by the uphill and downhill moves (an important capability for nested logit estimation).

The L function for Logit distribution is as follows for aggregate models:

$$Ln(L) = \sum_{(k,l)} \sum_i N_{kl} Ln(P_i) \quad (9)$$

Where N_{kl} is the number of trips made by the i th mode between the zones k and l , and P_i is as mentioned in equations 2.4 and 2.5. The aim is to estimate the vector of β' in P_i .

4. Simulated Annealing Algorithm

Simulated annealing's roots are in thermodynamics, where one studies a system's thermal energy. A description of the cooling of molten metal motivates this algorithm. After slow cooling (annealing), the metal arrives at a low energy state [4]. Inherent random fluctuations in energy allow the annealing system to escape local energy minima to achieve the global minimum. But if cooled very quickly (or "quenched"), it might not escape local energy minima and when fully cooled it may contain more energy than annealed metal. Simulated annealing attempts to minimize some analogue of energy in a manner similar to annealing to find the global minimum[2]. Details can be found in Press et al [9].

Early simulated annealing algorithms considered combinatorial systems, where the system's state depends on the configuration of variables. Perhaps the best known is the traveling salesman problem, in which one tries to find the minimum trip distance connecting a number of cities. Combinatorial simulated annealing has been used successfully in computer and circuit design [10, 11], pollution control [12], a special case of 0-1 programming [13], neural networks [14], reconstruction of Polly crystalline structures [15] and image processing [15].

Other global optimization algorithms have been introduced in recent years. They include adaptive random search [16], genetic algorithms [17], the filled function method [18], multi level methods [19] and a method using stochastic differential equations [20]. Both Vanderbilt et al. (1984) and Bohachevsky et al. (1986) have modified simulated annealing for continuous

variable problems. However, the Corana et al. implementation of simulated annealing for continuous variable problems appears to offer the best combination of ease of use and robustness, so it is used in this study. While a complete description can be found there, a summary follows. The essential starting parameters to maximize the function $f(X)$ are T , the initial temperature; X , the starting vector of parameters; and V , the step length for X . Note that X and V are both vectors of length n , the number of parameters of the model. Upper case refers to vectors and lower case to scalars (with the exception of temperature, T). A function evaluation is made at the starting point X and its value f is recorded. Next, a new X , X' , is chosen by varying element i of X ,

$$x'_i = x + r.v_i \quad (10)$$

Where r is a uniformly distributed random number from $[-1, 1]$ and v_i is element i of V . The function value f' is then computed. If f' is greater than f , X' is accepted, X is set to X' and the algorithm moves uphill. If this is the largest f' , it and X are recorded since this is the best current value of the optimum. If f' is less than or equal to f , the Metropolis criteria decides on acceptance (thermodynamics motivates this criteria). The value

$$P = e^{\left(\frac{f'-f}{T}\right)} \quad (11)$$

is computed and compared to p' , a uniformly distributed random number from $[0, 1]$.

S.A's major advantage over other methods is an ability to avoid becoming trapped at local minima. The algorithm employs a random search, which not only accepts changes that decrease objective function f , but also some changes that increase it. The latter are accepted with the above probability. If p is greater than p' , the new point is accepted and X is updated with X' and the algorithm moves downhill. Otherwise, X' is rejected. Two factors decrease the probability of a downhill move: lower temperatures and larger differences in the function's value. Also note that the decision on downhill moves contains a random element.

After N_s steps through all elements of X, (all such "N" parameters are set by the user) the step length vector SV is adjusted so that 50% of all moves are accepted. The goal here is to sample the function widely. If a greater percentage of points are accepted for x , then the relevant element of V is enlarged. For a given

temperature, this i increases the number of rejections and decreases the percentage of acceptances. After N times through the above T loops, the temperature, T, is reduced. The new temperature is given by

$$T' = r_t.T \quad (12)$$

Table 1. Simulated Annealing Algorithm for Maximization Problem

```

SET initial parameters and values
  Set C (controls how fast V adjusts)
  Set X (starting values for model parameters)
  Set V (should cover the entire range of interest in X)
  Set  $\varepsilon$  (convergence criteria)
  Set  $r_T$  (temperature reduction factor)
  Set  $T_0$  (initial temperature)
  Set N (# of times  $\varepsilon$  tolerance is achieved before termination)
  Set  $N_s$  (# of times through function before V adjustment)
  Set  $N_T$  (# times through  $N_s$  loop before T reduction)
CALCULATE f(X)
  X = X opt
  f = f opt
DO UNTIL convergence
  DO  $N_T$  times
    DO  $N_s$  times
      DO i = 1, n
         $x'_i = x + r.v_i$  {r is uniform random number on [-1,1]}
        CALCULATE f '(X')
        IF f'  $\leq$  f THEN
          apply Metropolis criteria
          IF accepted: X = X' & f = f'
        END IF
        IF f' > f THEN
          X = X' & f = f'
        END IF
        IF f' >  $f_{opt}$  THEN opt
          X = X', f = f',  $X_{opt} = X'$ , &  $f_{opt} = f'$ 
        END IF
      END DO
    END DO
    ADJUST V such that half of all trials are accepted
  END DO
IF change in  $f_{opt} < \varepsilon$  last N iterations &  $|f - f'| < \varepsilon$  THEN
  REPORT  $X_{opt}$ ,  $f_{opt}$  & V
  STOP
ELSE
  X =  $X_{opt}$  {start on current best optimum}
  T =  $r_t.T$  {reduce T}
END IF
CONTINUE

```

Where r is between 0 & 1. A lower temperature makes a given downhill move less likely, so the number of T rejections increase and the step lengths decline. In addition, the first point tried at the new temperature is the current optimum. The smaller steps and starting at the current optimum focuses attention on the most promising area.

The algorithm ends by comparing the last N values of the largest function values from the end of each temperature reduction with the most recent one and the optimum function value. If all these differences are less than ε , the algorithm terminates. This criterion helps ensure that the global maximum is reached.

The most important advantage of simulated annealing is that it can maximize functions that are difficult or impossible to otherwise optimize. The sole drawback to simulated annealing is the required computational power, but this problem has disappeared due to the continuing improvements occurring in high performance computing. Briefly, the power of the first Cray supercomputer can now easily be put on a desktop, and this trend shows no sign of slowing. Thus, simulated annealing is an attractive option for difficult functions [2].

5. Case Study for the Mode Choice Modeling of Isfahan

Isfahan is a city located in the central part of Iran. It's one of the five metropolitan cities in Iran and has a population of over 1.7 million and an area of about 178 km². There were two Comprehensive Studies of Transportation and Traffic conducted in Isfahan over the past 25 years. The recent one was on 2004 where the mode choice modeling of Isfahan was studied as a part of a Comprehensive Transportation Study of this city. In that study Sharif University of Technology used the GAUSS software for maximizing the likelihood function that used conventional algorithms for finding the parameters of Logit model in mode choice modeling [21] but these algorithms may be trapped in local minima.

In the following case study the same data base of the study mentioned (comprehensive studies of Transportation and Traffic of Isfahan 2004) is

used in order to compare the present study with a previous study conducted using a different estimation algorithm and showing the glowing points and superiority of S.A. The data base consisted of a 141 by 141 zone matrixes of different attributes used in mode choice modeling. They were collected by home surveys in 2001. The date of the data base is not important since this study is after proving the advantages of S.A over other optimization methods rather than an up to date result for the mode choice modeling of Isfahan and for the aim of this paper any same data base for the studies compared is acceptable and doesn't cause any error in optimization at all. Four modes of transport were used in this study for mode choice modeling, Bus, Taxi, Car, motorcycle-bicycle and work aimed trips were considered as it had the highest share comparing to other aimed trips(educational, shopping and etc) and had more sets of data available for use in modeling [21].

The parameters that build up the utility function in our model are as follows:

A_{co} = car ownership in the destination zone (obtained from the 2001 survey [21])

A_{mo} = motor ownership in the destination zone (obtained from the 2001 survey [21])

C_t = travel time by the taxi and auto-vehicle in minutes (obtained for all O-D from the trip Assignment model)

D_{st} = the shortest ground path between origin and destination in the road network in km.

T_{otx} = the time spent outside to find a taxi and going on board in minutes (8 minutes has been considered for this study [21])

T_{inb} = travel time inside the public transport vehicle (bus) in minutes. (Obtained for all O-D from the trip assignment model).

T_{out} =travel time outside the public transport vehicle (bus) in minutes. (Obtained for all O-D from the trip assignment model by adding the 2 quantities of walking and waiting time)

N_{brd} = number of times going on board using public transportation for going from an origin to a destination.

T_{tb} = the total travel time by public transport vehicle (bus) in minutes. (Obtained for all O-D By adding "T_{out}" and "T_{in}" [21]).

T_{tt} = the total travel time by taxi in minutes.

(Obtained for O-D from dividing D_{st} by 0.323[21])

T_{int} = travel time inside taxi in minutes. (Obtained for all O-D from the trip assignment model by subtracting T_{otx} (8min) from total travel time by taxi ($D_{st}/0.0323$)[21])

The study results estimated with different initial values and parameters and the result obtained from the Comprehensive studies of transportation and traffic [21] are shown in Table 2. All the coding and computation has been done using Matlab 7, and been run on a P4 processor with 1024 MB of RAM

In figure 1, X stands for the number of iterations and Y is the amount of LL function.

A statistic called the likelihood ratio index is often used with discrete choice models to measure how well the models fit the data.

The likelihood ratio index is defined as:

$$\rho = 1 - \frac{LL(\beta)}{LL(0)} \quad (13)$$

Where $LL(\beta)$ is the value of the log-likelihood function at the estimated parameters and $LL(0)$ is its value when all the parameters are set equal to zero. In comparing two models estimated on the

Table 2. Table of results showing the initial values used in programming and the parameters of Logit model for different cases

	Case 1	Case 2	Case 3	Case 4	Case 5	Case 6	Comprehensive studies of transportation results
	Final case						
T_0	40000	60000	60000	65000	65000	65000	-
Initial maximum step size	250	250	180	180	180	180	-
# results kept for checking convergence	16	20	20	20	20	20	-
ϵ	10	10	10	10	10	10	-
N_T	4	4	8	10	12	14	-
N_S	10	10	10	12	14	16	-
r_t	0.75	0.85	0.85	0.85	0.85	0.85	-
f_{opt} (Log-likelihood)	-3.0153e5	-2.9411e5	-2.8884e5	-2.8435e5	-2.8325e5	-2.8321e5	-2.9993e5
Final Temperature	0.3017	0.6878	0.3590	0.8766	0.4576	0.3890	-
Run time (sec)	138.54	229.85	473.64	680.9	1020.37	1330.35	-

		R_{opt}						
Car	α_c	-1.5785	-1.7745	-1.0019	-0.2737	-0.2347	-0.2419	-1.1350
	Aco	5.5122	14.1600	6.2539	9.4568	6.3714	5.7086	8.3100
	C_t	-0.0639	-0.0058	-0.0402	-0.0005	0	-0.0002	0
	C_t / D_{st}	-0.0649	-0.0610	-0.0011	-0.0001	-0.0005	-0.0002	-0.0140
Taxi	α_T	0	0	0	0	0	0	0
	$(D_{st} / 0.323) - Totx$	-0.0205	-0.0315	-0.0454	-0.0018	-0.0015	-0.0102	-0.0140
	$Totx / D_{st}$	-1.1521	-1.0422	-0.8233	-0.1284	-0.1336	-0.1574	-0.3510
	Aco	4.5884	11.6612	5.3926	6.1718	3.8416	3.2657	0
	$D_{st} / 0.323$	-0.0536	-0.0163	-0.0070	-0.0084	-0.0106	-0.0042	0
Bus	α_B	-2.0133	-1.2941	-1.815	-0.7204	-0.919	-0.8860	-0.7110
	Tin	-0.0385	0	-0.0010	0	0	0	-0.0140
	$Tout / D_{st}$	-0.1451	-0.0406	-0.0244	-0.0251	-0.0339	-0.0451	-0.2210
	$Ln(D_{st} / N_{brd})$	0.0677	0.1938	0.0422	0.1710	0.0791	0.0357	0.3230
	Tt	-0.0085	-0.0153	-0.0094	-0.0007	0	0	0
Motor+bike	α_M	-3.1354	-1.8669	-1.3089	-0.0744	-0.2199	-0.1856	-1.6270
	Amo	7.5351	14.7874	7.1580	9.1141	6.2692	5.6461	15.7460
	$D_{st} / 0.322$	-0.0031	-0.0041	-0.0207	-0.0090	-0.0085	-0.0118	0

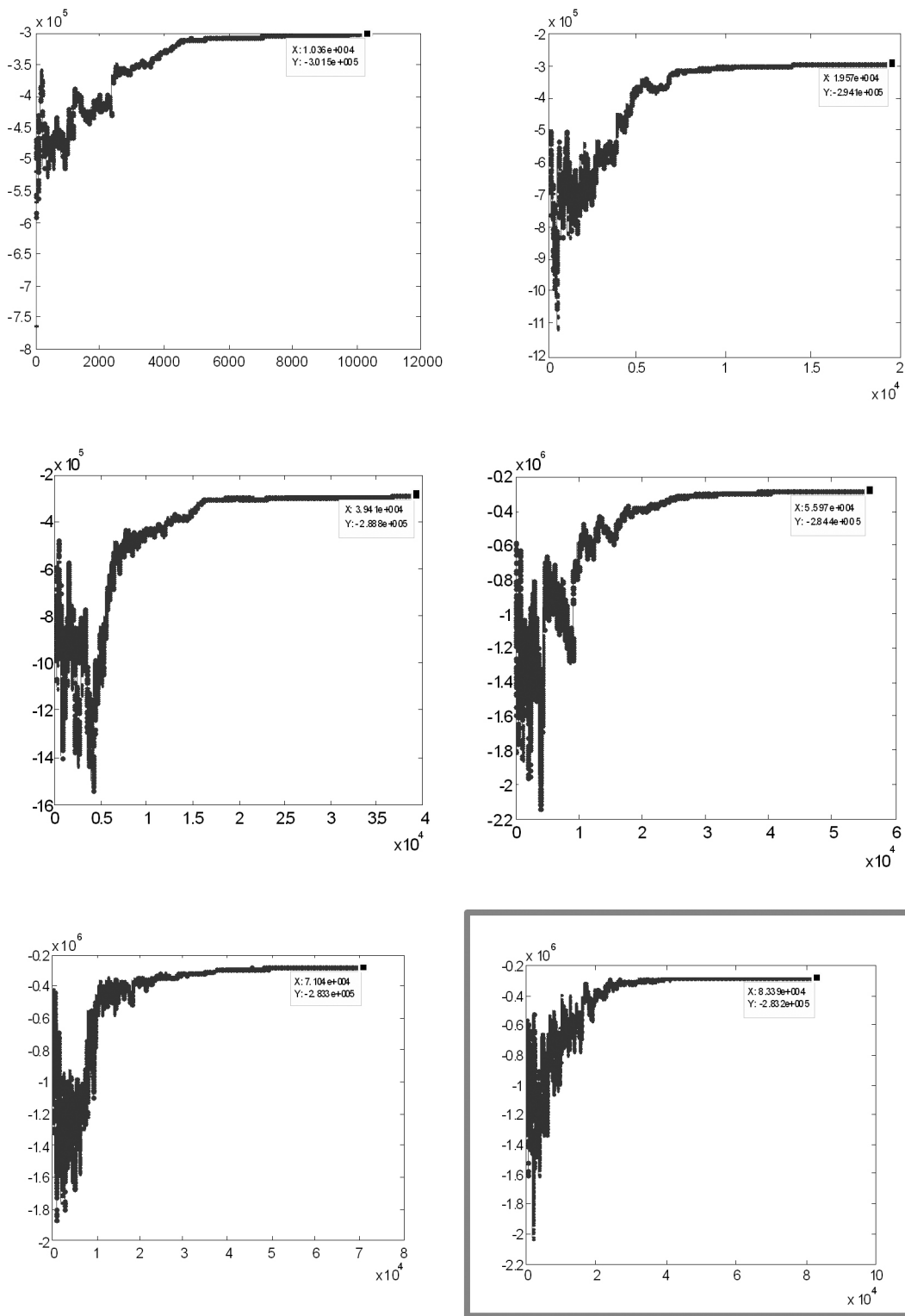


Fig. 1. SA approaches towards LL function maximum for cases

same data and with the same set of alternatives (such that $LL(0)$ is the same for both models), it is usually valid to say that the model with the

higher ρ fits the data better [1]. Table 3 shows the amount of ρ for our method comparing to the method used in the comprehensive transportation

study. We can see that both amounts are bigger in this study comparing to the results from the study of comprehensive studies of transportation in Isfahan. The results define the advantage and excellence of the method illustrated in this paper comparing to other algorithms for estimating the logit models.

6. Conclusion

Most conventional optimization algorithms cannot find the optimum value due to the zig zaging of the algorithms and that these algorithms terminate when the improvements in the objective function is not significant, a shortcoming that most heuristic algorithms have overcome and in this paper we have emphasized on this property and showed how it can improve the results.

Based on the results shown the proposed approach yields good results for a very important estimation problem. By evaluating the results for the final case (case 6) with the results from the

comprehensive studies of Transportation and Traffic (last column), we can see that the result obtained using S.A gives a greater maximum than that of the comprehensive study. This shows that the method mentioned is capable of finding the global optimum better than conventional algorithms used in software packages like GAUSS. In fact we could see that the result from the comprehensive studies is actually the local optimum rather than the global optimum which shows the power of S.A as it does not get trapped in local minima or maxima.

The widespread use of parameter estimation has turned it into a field that is of great interest to many practitioners and researchers. Something very interesting to research is a nice analysis on the SA algorithm itself with regard to the estimation problem for different discrete choice models and to establish a method to apply it for these models. This problem is a matter of finding the global optimum for these models that are used in different predictions that concern decision makers' choice. On the other hand as there are

Table 3. A comparison on the amounts of ρ and f_{opt} (Log-likelihood) for case 6 (final case of this study) and the results of the comprehensive transportation study

	Case 6(final case of this study)	Comprehensive studies of transportation
f_{opt} (Log-likelihood)	-2.8321E5	-2.9993E5
ρ	0.0955	0.04209

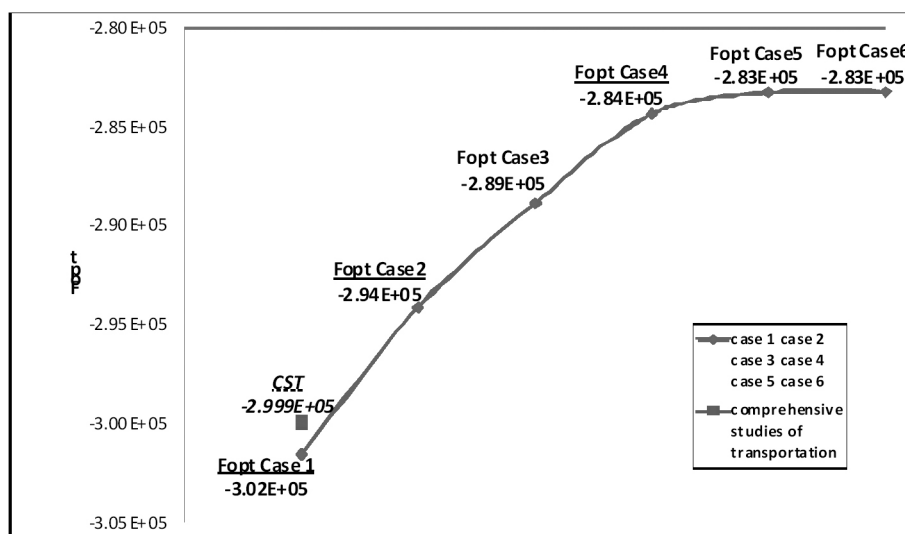


Fig. 2. f_{opt} (Log-likelihood) for different cases

other well-established meta-heuristics, such as genetic algorithms, Bee and Ant Colony, etc., a discussion of how good the estimation could use them, may lead to a better estimation. In addition a good subject for future research is to apply the method in this paper for estimation of Nested-Logit models (NL) that are more precise and eliminate the IIA problem regarding MNL models.

Acknowledgment

We would like to take this opportunity to thank Mr. Jalal Zahabi, from Yazd University, Department of Electrical Engineering, who helped us a lot on the programming and coding of our model in Matlab.

References

- [1] Kenneth E. Train, *Discrete Choice Methods with Simulation*, University of California, Berkeley and National Economic Research Associates, 2003
- [2] *Global Optimization of Statistical Functions with Simulated Annealing*, William L. Goffe, Gary D. Ferrier, John Rogers, 1993
- [3] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, E. Teller, *Equations of state calculations by fast computing machines*, *Journal of Chemical Physics* 21 (1958) 1087–1092.
- [4] S. Kirkpatrick, C.D. Gerlatt Jr., M.P. Vecchi, *Optimization by simulated annealing*, *Science* 220 (1983) 671–680.
- [5] R. Ross, *Graphical methods for plotting and evaluating Weibull distributed data*, in: *Proc. Int. Conf. Prop. Appl. Diel. Mater*, vol. 1, Brisbane, Australia, 1994, pp. 250–253.
- [6] E.H. Lehman, *Shapes, moments and estimators of the Weibull distribution*, *Transactions of IEEE on Reliability* 11 (1962) 32–38.
- [7] J.S. White, *The moments of log-Weibull order statistics*, *Technometrics* 11 (2) (1969) 374–386.
- [8] J.S. Usher, *Weibull component reliability-prediction in the presence of masked data*, *IEEE Transactions on Reliability* 45 (1996) 229–232.
- [9] Press, William H., Brian Flannery, Saul A. Teukolsky and William T. Vetterling. *Numerical Recipes, The Art of Scientific Computing*. New York: Cambridge University Press, 1986.
- [10] Kirkpatrick, S., C. D. Gelatt Jr. and M. P. Vecchi. "Optimization by Simulated Annealing." *Science* 220 (1983):671-80.
- [11] Wong, D. F., H. W. Leong and C. L. Liu. *Simulated Annealing for VLSI Design*. Boston: Kluwer Academic Publishers, 1988
- [12] Derwent, Dick. "A Better Way to Control Pollution." *Nature* 331 (1988): 575-78.
- [13] Drexl, A. "A Simulated Annealing Approach to the Multiconstraint Zero-One Knapsack Problem." *Computing* 40 (1988): 1-8.
- [14] Wasserman, Philip D. and Tom Schwartz. "Neural Networks, Part 2: What Are They and Why Is Everybody So Interested in Them Now?" *IEEE Expert Spring* 1988: 10-15.
- [15] Telly, H., Th. M. Liebling and A. Mocellin. "Reconstruction of Polycrystalline Structures: a New Application of Combinatorial Optimization." *Computing* 38 (1987): 1-11.
- [16] Pronzato, Luc, Eric Walter, Alain Venot and Jean-Francois Lebruchec. "A General-Purpose Optimizer: Implementation and Applications." *Mathematics and*

- Computers in Simulation 24 (1984): 412-22.
- [17] Goldberg, David E. Genetic Algorithms in Search, Optimization and Machine Learning. Reading MA: Addison-Wesley, 1989
- [18] Renpu, G. E. "A Filled Function Method for Finding a Global Minimizer of a Function of Several Variables," Mathematical Programming. 46 (1990): 191-204
- [19] Kan, A. H. G. Rinnooy and G. T. Timmer. "Stochastic Global Optimization Methods Part II: Multi Level Methods." Mathematical Programming. 39 (1987): 57-78.
- [20] Aluffi-Pentini, Filippo, Valerio Parisi and Francesco Zirilli. "A Global Optimization Algorithm Using Stochastic Differential Equations." ACM Transactions on Mathematical Software 14 (1988): 344-65.
- [21] Sharif University of Tech Research centre of Transportation, Comprehensive Studies of Transportation and Traffic of Isfahan, Sept 2004