ON THE CALIBRATION OF THE GRAVITY MODEL

Nanne J. van der Zijpp[1]
Transportation and Traffic Engineering Section
Delft University

Benjamin G. Heydecker
Centre for Transport Studies
University College London

1.   INTRODUCTION

Knowledge of Origin-Destination (OD) trip matrices is needed in many stages of transport planning. As direct observations of OD-flows are usually not available, a commonly used approach is to estimate them from traffic counts and other aggregated flow constraints. The under-specification that characterizes this estimation process can be resolved by requiring that OD-values conform to a model of travel demand. Several approaches of this kind have been discussed in literature and are being used in practice. In this paper we consider a class of methods based on the gravity model, with either an exponential or a piecewise constant deterrence function. We focus on numerical methods to calibrate these models from traffic counts and establish a unifying analysis of them.

The paper starts with a description of the gravity model, and a summary of the assumed modelling and observation error structures. Subsequently a likelihood expression is established which we seek to maximise. The paper then concentrates on methods that can be used for this purpose. Existing methods are discussed, and it is shown that some of the difficulties that occur when applying them can be resolved by a change of variables to logarithms of the model parameters. A key result concerns the matrix of second derivatives of the likelihood function. This result is then applied and leads to a new optimisation method. In a series of experiments this method is shown to be robust and computationally efficient.

2.   MODELLING

2.1   Model Description

The description of the gravity model that is used in the present paper is given by:

$$T_{ij} = T_{ij}^{m} + \varepsilon_{ij}$$

$$T_{ij}^{m} = O_i D_j \, F(c_{ij}),$$
$$i=1,2,\ldots I, \ j=1,2,\ldots J, \tag{1}$$

---

1. A large part of the work on which the paper is based was performed while the first author worked at the Center for Transport Studies, University College London

with:

T_{ij}	OD-flow from zone i to zone j
I,J	number of origins, number of destinations
T_{ij}^m	model estimated OD-flow
ε_{ij}	model residual
O_i	production ability
D_j	attraction ability
$F(.)$	deterrence function
c_{ij}	generalized travel costs

A plausible approach is to specify the conditional distribution of the OD-flows, given the model estimated OD-flow as a Poisson distribution, with the model estimated OD-flow as its expectation. However, for analytical convenience we adopt a normal distribution for this instead, with variance equal to expectation, noting that for sufficiently aggregated OD-tables (e.g. $T_{ij} > 8$) this only represents a minor change. This is effected by the assumption:

$$\varepsilon_{ij} \sim N[0, T_{ij}^m] \tag{2}$$

Various functional forms for the deterrence function are found in literature, see e.g. *Ortuzar and Willumsen (1990)* for an overview. In the present paper we will confine ourselves to two popular cases:

Case 1: The exponential deterrence function

$$F(c_{ij}) = \exp[-\alpha\, c_{ij}] \tag{3}$$

Case 2: The piecewise constant deterrence function

$$F(c_{ij}) = \sum_{h=1}^{H} F_h\, \delta_h(c_{ij}) \tag{4}$$

where:

H	number of cost ranges
$\delta_h(c_{ij})=$	$\begin{cases} 1, \text{ if } c_{ij} \text{ is in cost range } h. \\ \\ 0, \text{ otherwise.} \end{cases}$

It is assumed that each OD-pair has costs in exactly one range. The piecewise constant function may be thought of as a histogram, in which the height of each bar corresponds to the deterrence value for travel costs within the corresponding cost range.

2.2 Traffic Counts

The model parameters are to be estimated from traffic counts or other aggregated flow observations. In our analysis, traffic counts y_k are assumed to be arbitrary linear combinations of OD-flows:

$$y_k = \sum_{i=1}^{I} \sum_{j=1}^{J} T_{ij}\tau_{ijk} + \xi_k \qquad k=1,2,\ldots K \tag{5}$$

with:

K	number of traffic counts
y_k	traffic count at location k
ξ_k	counting error at location k
τ_{ijk}	proportion of flow from i to j that traverses link k

The assignment proportions τ_{ijk} are assumed to be given. The physical counting errors ξ_k are assumed to be zero mean, mutually independent and normally distributed, with known variance ω_k^2. Model estimated traffic counts y_k^m are defined by substituting model estimated OD-flows T_{ij}^m in the r.h.s. of (5). If needed, the approach presented in this paper may be made more general by allowing for the use of a prior matrix. In such case, each cell would correspond to a separate further observation.

2.3 Maximum Likelihood Approach to Model Calibration

The following notation summarizes the symbols that were introduced earlier as vectors and matrices, and will be used in the derivation of a maximum likelihood (ML) estimator for the model parameters: their sizes are implicit in their definitions.

O, D, F	vectors of production abilities, attraction abilities and deterrence function parameter(s) respectively.
$T, x(i,j)$	vector of OD-flows, location of matrix element T_{ij} in T.
y	vector of traffic counts
ξ	vector of counting errors
T^m, y^m	vectors of model estimated OD-flows and model estimated traffic counts respectively
$\boldsymbol{T^m}$	diagonal matrix with the elements of T^m on its main diagonal.
Ω	covariance matrix associated with the counting errors, which as a result of earlier assumptions is diagonal.
τ	matrix of assignment proportions, $\tau_{k,x(i,j)} = \tau_{ijk}$, $k=1,2,...K$

Using this notation, (5) is summarized as:

$$y = \tau\, T + \xi \tag{6}$$

Furthermore, the equivalent of (1) and (2) in matrix notation is:

$$T \sim \text{MVN}[T^m, \boldsymbol{T^m}] \tag{7}$$

i.e. the distribution of T is given by a multivariate normal distribution with expected value T^m, and covariance matrix $\boldsymbol{T^m}$. Combining these two equations results in an expression for the conditional distribution of the traffic counts given the model parameters:

$$y \sim \text{MVN}(\tau\, T^m, \tau\, \boldsymbol{T^m}\, \tau' + \Omega) \tag{8}$$

where T^m should satisfy the requirements that follow from (1). In theory, the parameters O, D and F can be found by maximising the joint likelihood, so that these parameters are given by:

$$\begin{bmatrix} O \\ D \\ F \end{bmatrix} = \begin{matrix} \text{argmax} \\ O, D, F \end{matrix} \ \text{MVN}(\tau \ T^m, \tau \ \boldsymbol{T^m} \ \tau' + \Omega) \ \Big|_y \qquad (9)$$

2.4 Simplifying the Likelihood Objective Function

In (9) both the expectation and the covariance matrix of the density that is to be max-imised depend on the model parameters. It seems that any algorithm designed to solve (9) would do so in an indirect manner, for example by keeping the covariance matrix constant while solving the model parameters, and then updates the covariance matrix. Such a process could continue until a fixed point is found, as is suggested in figure 1. A covariance matrix that could be used to initialize such an iterative scheme, would be given by a matrix with the elements of y on the main diagonal that account for model residuals (see eq.(2)), increased with the covariance matrix of physical counting errors. This matrix will be denoted by $Y+\Omega$. This iterative process is expected to converge as the solution of each individual step in this process is not especially sensitive to the cov-ariance matrix that is used. However, the present paper does not provide a mathemati-cal proof of convergence.

In the remainder of this paper, we discuss algorithms that can be used to maximise the likelihood if the covariance matrix is fixed. This corresponds to solving the following optimisation problem:

$$\begin{matrix} \text{minimize} \\ O \geq 0, \ D \geq 0, \ F \geq 0 \end{matrix} \ J(O,D,F) = (\ y\text{-}\tau \ T^m\ )' \ \Sigma^{-1} \ (\ y\text{-}\tau \ T^m\ )$$

$$T^m = T^m(O,D,F) \qquad (10)$$

Where Σ is the covariance matrix that may or may not be updated during the process. There are several advantages in formulating the OD-estimation in this way, compared to formulations derived from a maximum entropy arguments. The main advantages are:

- The traffic counts that are used may represent arbitrary linear combinations of OD-cells,
- The objective function offers the possibility to calibrate a gravity model, even if no explicit data are available on trip ends and/or trip length distribution,
- The possibility exists to specify the variance of counting errors, or to weigh the observations according to the confidence in them,
- Spatial dependencies that occur due to the fact that the model prediction error, ε_{ij}, is observed at several locations can be accommodated by using the iterative scheme shown in figure 1.

The objective function (10) is a generalization of approaches that are used in practice, e.g. Willumsen (1991).

3.   EXISTING ALGORITHMS FOR CALIBRATING THE GRAVITY MODEL

Note that (10) describes several classes of optimisation problems, depending on the form of the deterrence function. Here we consider separately the exponential deterrence function (case 1, see equation (3)) and the piecewise constant deterrence function(case 2, see equation (4)). For each of these classes of problem, special algorithms exist.

3.1   Balancing Methods

The *bi-proportional method* starts with an initial or prior matrix. This matrix is subsequently scaled row by row and then column by column to yield the observed or given trip end totals. Therefore, after each iteration the resulting matrix can be written as:

$$T_{ij}^m = \prod_{k=1}^{K} \alpha_{ik} \prod_{k=1}^{K} \beta_{jk} T_{ij}^p \tag{11}$$

where:

α_{ik} multiplicative factor applied to row i in the kth iteration.

β_{jk} multiplicative factor applied to column j in the kth iteration.

T_{ij}^p prior matrix

K number of iterations

The iterations stop if no more substantial changes occur in the estimated matrix, i.e. when the observations are reproduced exactly. An equivalence with the gravity model with exponential deterrence function exists if T_{ij}^p is replaced by exp[-α c_{ij}]. In that case equation (11) can be summarized as:

$$T_{ij}^m = O_i D_j \exp[-\alpha\, c_{ij}] \tag{12}$$

Note that the bi-proportional method can not be used to find appropriate values for α. For this purpose Hyman's method (Hyman, 1969) can be used. A prerequisite to apply the latter method is that an observed trip length distribution (OTLD) is available.

The *tri-proportional method* implies that in addition to this, the prior matrix is scaled to force the estimated matrix to match an OTLD. In this case the estimated matrix can be written as:

$$T_{ij}^m = \prod_{k=1}^{K} \alpha_{ik} \prod_{k=1}^{K} \beta_{jk} \left[\sum_{h=1}^{H} \delta_h (c_{ij}) \prod_{k=1}^{K} \gamma_{hk} \right] T_{ij}^p \tag{13}$$

where:

γ_{hk} multiplicative factor applied during the kth iteration to OD-cells with corresponding costs, c_{ij}, in cost bin h.

$\delta_h(c_{ij})$ membership function as defined in equation (4).

An equivalence with the gravity model with piecewise constant deterrence function is

reached if the prior matrix is replaced with a matrix of ones. In that case (13) simplifies to:

$$T_{ij}^m = O_i D_j \sum_{h=1}^{H} F_h \, \delta_h(c_{ij}) \qquad (14)$$

In a balancing method, the iterations continue until all observations are met by the estimated values. A requirement for convergence of the bi- and tri-proportional balancing methods is that such a solution exists, i.e. that the traffic counts that are used are mutually consistent. The analogy between balancing methods and other gravity model calibrating methods becomes weaker if an initial matrix is scaled to meet measurements that do not represent trip ends, as is done in entropy maximizing approaches such as ME2 (Ortúzar and Willumsen, 1990; Willumsen, 1991; Van Zuylen and Willumsen, 1980). Similarly, if an arbitrary prior matrix T_{ij}^p is used, the solution resulting from applying a balancing method can not be written in the forms (12) or (14). Balancing methods can hence only be applied to find a candidate solution to (10) in cases where:
- observations and constraints represent trip ends or observed trip length distributions, and
- observations are mutually consistent.

In this case, the solution that will be found will exactly reproduce the observations and will hence correspond to a zero value for the objective function (10), which also is the global minimum. Balancing methods are popular due to their ease of implementation. However, convergence can be slow (see Maher, 1983) if not uncertain. Proofs of convergence only apply to special cases, e.g. only trip-ends are given and the deterrence function is kept constant, see Evans (1971) and Kirby (1974). In the more general case where observations represent traffic counts, and may not be mutually consistent, other techniques should be considered, two of which are discussed below.

3.2 Gradient Search Methods

With some effort it is possible to establish the derivatives of the objective function (10) with respect to the model parameters O_i, D_j and F_h. The resulting gradient vector can be used as a basis for an iterative scheme to minimise (10). The simplest way this can be done is to apply a steepest descent method. However, steepest descent method tend to converge very slow. A gradient search methods with a higher speed of convergence is the conjugate gradient method, see Bazaraa et al. (1993). An easy way to find conjugate gradient search directions was proposed in Fletcher and Reeves (1964). There are however a number of difficulties that occur when a gradient search method is applied to minimise (10):
- The solution to the line searches can not be expressed in analytical way. Hence, the line searches must be done numerically,
- The parameter space is constrained to non-negative values. Again this makes the line search more complex, and removes the theoretical basis of conjugate gradient methods,
- It is difficult to establish an effective convergence criterion,
- Convergence can be extremely slow.

These factors make it difficult to construct a gradient search method that operates in a

reliable way in all circumstances.

3.3   Gauss-Seidel Methods

Another way to tackle the minimization problem (10) is to divide the model parameters in groups and to solve the parameters in each group separately, while keeping the parameters in other groups temporarily constant. This iterative process is continued until a fixed point is found. For problem (10) it seems advantageous to first solve the vector of production abilities (O), then solve the vector of attraction abilities (D), and then solve the vector of deterrence function values (F) (this procedure can only be used to calibrate a gravity model with a piecewise constant deterrence function). Each step in this sequence corresponds to minimizing a quadratic function. If the nonnegativity constraints can be ignored, this can be solved at the expense of one matrix inversion, so line searches can be avoided. The details of the method are described in Van Der Zijpp and de Romph (1996).

Although the method works well in a large number of cases and is easy to implement with the help of modern software packages, there are a number of technical difficulties:
- No theoretical proof of convergence exists,
- The minimization problems can be expensive to solve if the nonnegativity constraints become active,
- The method can not be applied to calibrate gravity models with exponential deterrence functions,
- It is difficult to establish an effective convergence criterion.

4.   A UNIFYING ANALYSIS OF THE ML OBJECTIVE FUNCTION

We now show how, by using a logarithmic transformation, the gravity model formulations presented here can be unified and solved conveniently.

4.1   The Logarithmic Transformation

It has been shown earlier (see e.g. Sen and Smith, 1995) that both the gravity model with exponential deterrence function (equations (1) and (3)), and the gravity model with piecewise constant deterrence function (equations (1) and (4)) can be summarized in the following form:

$$t=\exp(Cq) \tag{15}$$

where

$\quad\quad\quad$ exp(.)$\quad\quad$ is the exponentiation operator, applied element by element

$\quad\quad\quad$ q $\quad\quad\quad\quad$ is the vector of logarithms of the model parameters, i.e.

$$q = \log\left(\begin{bmatrix} O \\ D \\ F \end{bmatrix}\right)$$

$\quad\quad\quad$ C $\quad\quad\quad\quad$ is a matrix with three nonzero elements per row. The first and second nonzero elements are given by respectively:

$$C_{x(i,j),\,i} = 1$$

$$C_{x(i,j),\,I+j} = 1 \tag{16}$$

The way the third nonzero element is defined depends on whether an exponential or a piecewise constant deterrence function is used:

Case 1: The exponential deterrence function

$$C_{x(i,j),\,I+J+1} = -c_{ij} \tag{17}$$

Case 2: The piecewise constant deterrence function

$$C_{x(i,j),\,I+J+h} = \delta_{hij} \quad , \quad h=1,2,\ldots H \tag{18}$$

In summary, the problem of estimating OD-matrices from traffic counts using a gravity model combined with either an exponential or a piecewise constant deterrence function is equivalent to the minimization of the following objective function:

$$J(q)=(y\text{-}At(q))'\, \Sigma^{-1}\, (y\text{-}At(q)),$$
$$t=\exp(Cq) \tag{19}$$

Compared to the formulation given in (10) this change of variables has the following practical advantages:
- A single objective function can be used to represent methods based on exponential and piecewise constant deterrence functions,
- The minimization is not subject to nonnegativity constraints. This argument does not apply strictly in case of the exponential deterrence function, as one can argue that it would not be realistic if the parameter α in the deterrence function (3) assumed negative values,
- Ease of notation and analysis. This advantage will become apparent in the next section, where expressions for the gradient and the Hessian (the matrix of second derivatives) of the objective function with respect to the vector q will be given.

4.2   Deriving the Gradient and Hessian of the Objective Function

Applying some analysis to the objective function (19) has led to expressions for the gradient and the Hessian (the matrix of second derivatives) of this function. These are given respectively by:

$$\nabla_q[\,J\,]=\left[\frac{\partial J}{\partial q_i}\right]=2C'\mathrm{diag}(t)A'\,\Sigma^{-1}\,(At\text{-}y) \tag{20}$$

$$\nabla^2{}_q[\,J\,]=\left[\frac{\partial^2 J}{\partial q_i\partial q_j}\right]=2C'[\mathrm{diag}(t)\mathrm{diag}(A'\,\Sigma^{-1}\,(At\text{-}y))+\mathrm{diag}(t)A'\Sigma^{-1}A\,\mathrm{diag}(t)]C \tag{21}$$

In these equations, the diag(.) operator transforms a vector into a diagonal matrix. For a full derivation of (20) and (21) we refer to Van Der Zijpp and Heydecker (1996b). Equations (20) and (21) give insight into the performance of existing solution methods, and provide a foundation for new ones.

4.3 Implications for Existing Methods

The gradient search methods and Gauss-Seidel methods discussed in section 3 can be used effectively to find stationary points of an objective function. Whether such a point represents a local minimum or a saddle point can easily be determined by evaluating the Hessian at that point: a local minimum only exists if the Hessian is PD. Therefore result (21) can be used as a useful addition to existing methods: it supplies a powerful convergence criterion. However the analysis needed to determine if a particular local minimum to objective function (19) is also a global minimum is more intricate; this problem is still open to investigation.

5. A NEW ALGORITHM TO CALIBRATE GRAVITY MODELS

Using expression (21) for the second derivatives, it is possible to define a new numerical method to minimize (19). This method is based on a quadratic approximation of the objective function of the form:

$$J^*(q+\varepsilon)=J(q) \; + \; \varepsilon'.\nabla_q[\,J(q)\,] \; + \; \tfrac{1}{2}\,\varepsilon'\,\nabla^2_q[\,J(q)\,]\,\varepsilon \qquad (22)$$

This expression can be minimised over the vector ε at the cost of one matrix inversion, provided that the Hessian of J in point q is Positive Definite (PD). An iterative method based on this strategy is known as the Newton method. The advantage of this method is that each successive iterate can be found analytically, so no numerical line search procedure is needed. Two factors should be taken into account when applying this method to minimize objective function (19):

- The Hessian $\nabla^2_q[\,J(q)\,]$ need not necessarily be strictly PD for all q. Therefore the quadratic approximation of the objective function, J^*, may not always be bounded below. In such cases the next iterate would not be defined. Therefore a modified Newton will be used to ensure that each successive iterate exists and will reduce the objective function. In the present paper, we adopt the Levenberg-Marquardt type of method, see Bazaraa et al. (1993), pp. 312. This method will be explained below.
- The vector q contains the logarithms of the model parameters. This ensures that the nonnegativity requirements are satisfied. However there does not exists a bounded value of q that corresponds to a solution to (9) that contains one or more zero parameter values. Although the value of zero can be approximated arbitrarily closely, it may not be possible to represent such small numbers internally in a computer, resulting in problems of convergence caused by rounding errors or underflow. Therefore the following penalty function is added to the objective function:

$$f(q) = c_1 \sum_i \max\,(0, -c_2 - q_i)^3 \qquad (23)$$

This function is continuous in its first and second derivatives, and prevents the above mentioned problems. The values that were chosen for c_1 and c_2 were 100 and 20 respectively.

The idea behind the Levenberg-Marquardt method is that even if the Hessian in a particular point is not PD, a next iterate with a lower objective function value can be found by performing a small step in the direction of the negative gradient. This is

effected by adding a multiple of the identity matrix, i.e. $\mu \mathbf{I}$, to the Hessian in (22), before minimizing this expression. The values of μ are modified between iterations using a strategy that is illustrated in figure 2. The next value of μ depends strongly on the ratio R of the true improvement and the predicted improvement. If this ratio is favourable then the value is μ is decreased, if this ratio is poor the value of μ is increased. The thresholds that were used are given in Bazaraa et al. (1993).

6.   NUMERICAL EXPERIMENTS

Numerical experiments have been performed on a set of empirical trip data obtained from a French toll road. These data consist of an observed trip table for eight different days between 29 origins and 26 destinations, resulting in 570 OD-flows that contribute to link-flows observed in the study area. There is no route choice in this network, and 107 traffic counts have been synthesized from these data. Two types of experiments were undertaken.

The first series of experiments involved estimating an OD-matrix by calibrating a gravity model on the traffic counts. This was done for a model with a piecewise constant deterrence function, see eq. (4). The number of distance classes that were distinguished was varied between 1 and 16. Each additional class was formed by subdividing an existing one, so that a hierarchical sequence of models was constructed.

From the models discussed in this paper, each of the conjugate gradient, Gauss-Seidel and modified Newton method were implemented. The conjugate gradient search method turned out to be extremely slow, which prevented the completion of a full set of experiments for this method. In table 1 the results were summarized for the Gauss-Seidel method and the modified Newton method. Each row in this table contains the results of calibrating the model using a deterrence function with the corresponding number (N) of classes, averaged over the eight different days for which observed data were available. The model prediction error could be computed by comparing the model estimated OD-flow with the value contained in the toll-ticket database, and is expressed using the Root Mean Squared Error (RMSE). The log-likelihood value is denoted with L, and decreases with the addition of each extra class. The other columns in the table contain the average CPU-time (on a Pentium 90) and the number of cases for which the algorithm did not converge before a preset maximum number of 2000 iterations was reached (indicated with the symbol E).

In all cases, a solution with an equal or higher likelihood value was computed with the modified Newton method. Moreover, with this method a solution that satisfied the convergence criterion was reached without exception, whilst the Gauss-Seidel exceeded the maximum number of iterations in 12 of the 128 instances. The main advantage of the modified Newton method is its speed: none of the cases that were considered took more than 290 seconds, whilst the Gauss-Seidel consumed 2975 seconds in one case; the average values were 106 and 626 seconds respectively. The experiments also revealed that for this particular dataset the best fit according to the RMSE measure is obtained if a deterrence function with six classes is used. Unfortunately, in practice it is not possible to determine this number as one needs a completely observed trip matrix to compute this measure. This issue is discussed in Van Der Zijpp and Heydecker (1996a).

In a second series of experiments each of the 570 observed OD-flows is used as a measurement. This is similar to the case in which a prior matrix is available. When calibrating the gravity model on these data the solution with the highest likelihood value and that with the lowest RMSE coincide: the resulting RMSE is known as the *model specification error*, see Van Der Zijpp and Heydecker (1996a). The results of the calibration are shown in table 2. On average, the computation time needed for the Gauss-Seidel method has increased, whilst the computation time needed for the modified Newton method has decreased substantially. Again the Newton method converged in all cases, whilst the Gauss-Seidel did not meet the convergence criterion in 19 of the 128 computations. The maxima of the computation times over these 128 computations are 4671 seconds for the Gauss-Seidel method and 16.42 seconds for the Newton method; the average values were 1489 and 9.82 seconds respectively.

7. CONCLUSION

It has been shown that the problem of calibrating a gravity model is equivalent to the minimization of a highly non-linear objective function. This formulation has a number of advantages over more traditional ones, including the possibility to use arbitrary traffic counts and constraints, not requiring the presence of an observed trip length distribution (OTLD) and taking into account counting errors and spatial dependencies. However, the minimization of this objective function is quite complex. Specific difficulties are the presence of the nonnegativity constraints, slow convergence, the absence of an adequate convergence criterion, and the complexity of implementation. The problem of the nonnegativity constraints can be avoided by a change of variables to the logarithms of the model parameters. This formulation unites methods with exponential and piecewise constant deterrence functions. A key result presented in the paper is an expression for the matrix of second derivatives of the objective function. Based on this matrix a minimization method is presented with a greatly improved speed of convergence. Moreover, this matrix, which can be computed easily with the aid of modern software packages, provides an effective convergence criterion that can also be applied in combination with other solution methods, such as Gauss-Seidel and gradient search methods.

8. REFERENCES

Bazaraa, M.S., Sherali, H.D. and Shetty, C.M. (1993) *Nonlinear Programming, Theory and Algorithms*, second edition, Wiley

Evans, A.W. (1971) The Calibration of Trip Distribution Models with Exponential or Similar Cost Functions, *Transportation Research*, Vol. 5, pp. 15-38

Fletcher, R. and Reeves, C.M. (1964) Function Minimization by Conjugate Gradients, *The Comp. J.*, 7, pp. 149-154

Hyman (1969) The Calibration of Trip Distribution Models, *Environment and Planning*, 1(3), pp.105-112

Kirby, H.R. (1974) Theoretical Requirements for Calibrating Gravity Models, *Transportation Research*, Vol. 8. pp. 97-104

Maher, M.J. (1983) Speeding up the Furness Method, *Traffic Engineering and Control*, pp.485-486L

Ortúzar, J. D. and Willumsen, L.G. (1990) *Modelling Transport*, Second Edition, JohnWiley and Sons, Chichester, England, ISBN 0 471 94193 X

Willumsen, L.G. in: Papageorgiou, M. (1991) *Concise Encyclopedia of Traffic & Transportation Systems*, Pergamon Press, pp.317

Sen, A, and Smith, T.O. (1995) *Gravity Models of Spatial Interaction Behavior*, Springer, ISBN 3-540-60026-4

Van Der Zijpp, N.J. and Heydecker, B. (1996) How many Parameters should a Traffic Model have?, *UTSG, ann.conference*, University of Huddersfield, unpublished

Van Der Zijpp, N.J. and Heydecker, B. (1996) Improved Algorithms for Calibrating Gravity Models *4th Meeting of the EURO Working Group on Transportation*, to be held at the University of Newcastle upon Tyne, 9-11 September 1996

Van Der Zijpp, N.J. and de Romph E. (1996) A Dynamic Traffic Forecasting Application on the Amsterdam Beltway, *Int. Journal of Forecasting*, 1996, forthcoming

Van Zuylen, H.J. and Willumsen, L.G. (1980) The most Likely Trip Matrix Estimated from Traffic Counts *Transportation Research-B*, Vol. 14B, pp. 281-293

Table 1: Estimation results based on traffic counts
for the number of classes ranging from 1 to 16

	Gauss-Seidel				Newton			
N	RMSE	L	av.CPU	E	RMSE	L	av.CPU	E
1	537.33	-17443.93	321		536.91	-17443.90	77	
2	485.57	-11308.71	245		486.09	-11308.68	223	
3	470.82	-8184.56	255		470.81	-8184.55	151	
4	478.93	-7514.64	984		480.46	-7182.98	86	
5	387.47	-5176.33	506		387.45	-5176.31	166	
6	380.23	-4940.86	202		380.18	-4940.86	61	
7	381.40	-4934.20	113		381.46	-4934.19	37	
8	639.94	-4539.26	1181	3	643.58	-4538.52	94	
9	805.85	-4229.48	1648	5	806.40	-4227.91	64	
10	783.25	-3533.62	1099	2	784.37	-3533.28	150	
11	840.49	-3467.37	634		839.90	-3467.30	61	
12	840.47	-3467.37	56		839.90	-3467.30	33	
13	843.22	-3459.97	436		843.65	-3459.91	94	
14	872.92	-3427.43	439		874.67	-3427.38	51	
15	700.29	-2999.34	724		699.89	-2998.97	137	
16	777.60	-2719.70	1176	2	777.74	-2719.48	206	
	tot.av.CPU		626				106	
	max. CPU		2975				289	

Table 2: RMSE fit of the gravity model
for the number of classes ranging from 1 to 16

	Gauss-Seidel				Newton		
N	RMSE	av.CPU	E		RMSE	av.CPU	E
1	329.36	307			329.36	14.82	
2	273.12	408			273.12	10.29	
3	235.05	946			235.05	11.08	
4	224.08	769			224.08	9.75	
5	216.94	807			216.94	10.38	
6	215.03	1191			215.03	8.23	
7	214.74	1392	1		214.74	7.83	
8	213.83	837			213.83	7.74	
9	212.45	1055			212.45	10.80	
10	212.37	1093	1		211.28	8.86	
11	207.30	1664	2		206.71	9.25	
12	205.97	1697	1		205.87	8.63	
13	203.40	1891	2		203.20	9.20	
14	202.53	2422	2		202.49	8.33	
15	196.48	4165	7		196.35	13.13	
16	196.23	3180	3		196.22	8.82	
	tot.av.CPU	1489				9.82	
	max. CPU	4671				16.42	
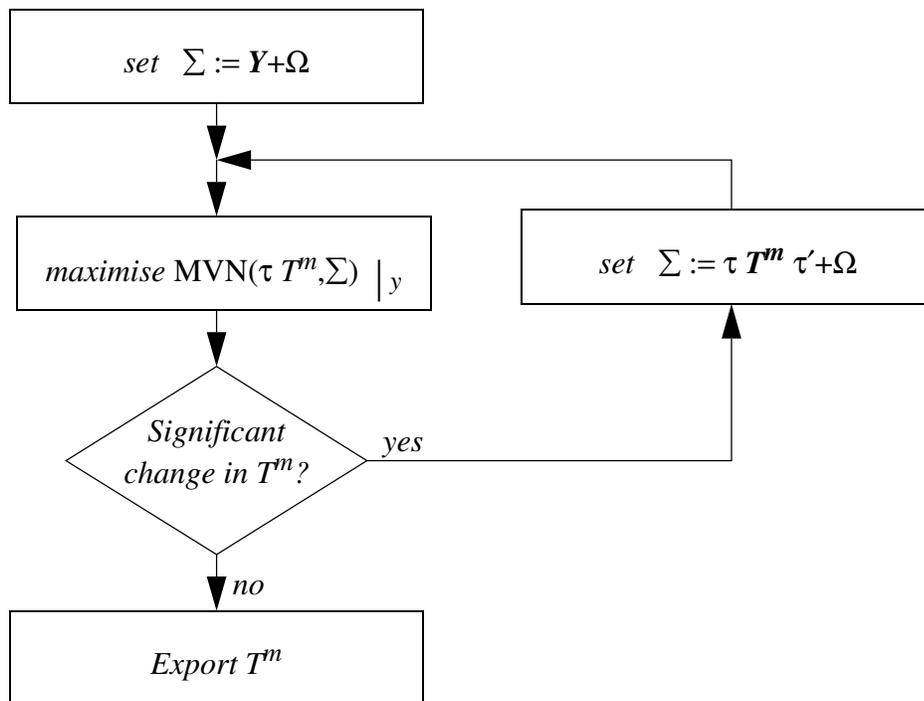
Figure 1:   *Outline of iterative process to maximise expression (8)*

initialize:
$k:=1,\ q^{(k)}:=\mathbf{1},\ \mu:=1$

$H = \left[\nabla^2_q[\,J(q^{(k)})\,] + \mu\,\mathbf{I}\right]$

H is P.D. — no → $\mu := 4\mu$

yes

$q^{(k+1)} := \underset{q}{\mathrm{argmin}}\, J^*(q)$

convergence reached?

no

$R:=[J(q^{(k)})-J(q^{(k+1)})]/[J^*(q^{(k)})-J^*(q^{(k+1)})]$

$R>0.75?$ — yes → $\mu := \mu/2$

no

$R<0.25?$ — yes → $\mu := 4\mu$

no

$R<0?$ — yes → $q^{(k+1)} := q^{(k)}$

no

$k = k+1$

export $q^{(k+1)}$
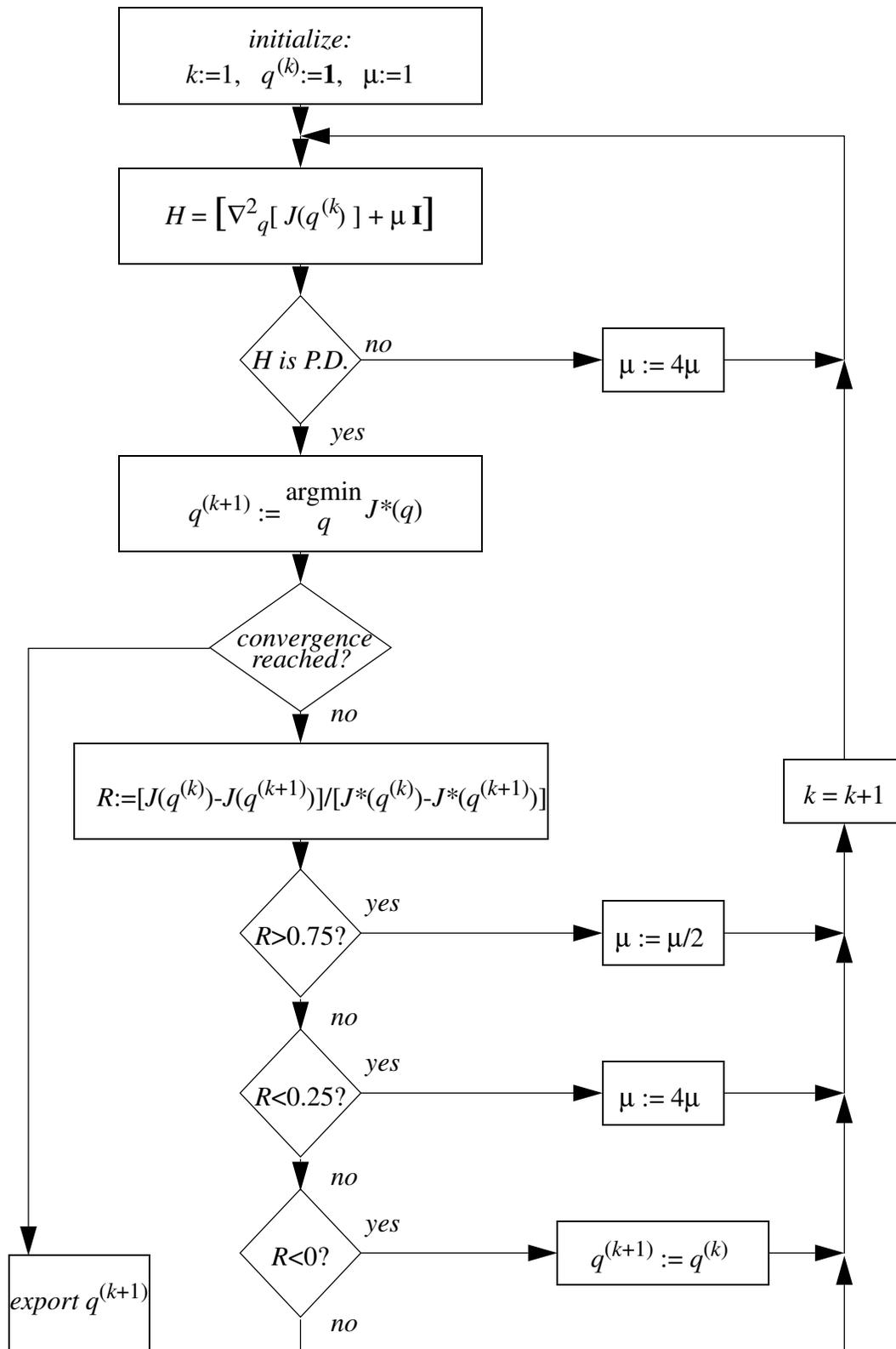
Figure 2:   Overview of Levenberg-Marquardt method (modified Newton method)

ON THE CALIBRATION OF THE GRAVITY MODEL

Nanne J. van der Zijpp
Transportation and Traffic Engineering Section
Delft University

Benjamin G. Heydecker
Centre for Transport Studies
University College London

ABSTRACT

Knowledge of Origin-Destination (OD) trip matrices is needed in many stages of transport planning. As direct observations of OD-flows are usually not available, a commonly used approach is to estimate them from traffic counts and other aggregated flow constraints. The under-specification that characterizes this estimation process can be resolved by requiring that OD-values conform to a model of travel demand. In this paper we consider a class of methods based on the gravity model, with either an exponential or a piecewise constant deterrence function. We focus on numerical methods to calibrate these models from traffic counts and establish a unifying analysis of them. The paper starts with a description of the gravity model, and a summary of the assumed modelling and observation error structures. Subsequently a likelihood expression is established which we seek to maximise. The paper then concentrates on methods that can be used for this purpose. Existing methods are discussed, and it is shown that some of the difficulties that occur when applying them can be resolved by a change of variables to logarithms of the model parameters. A key result concerns the matrix of second derivatives of the likelihood function. This result is then applied and leads to a new optimisation method. In a series of experiments this method is shown to be robust and computationally efficient.