Separation Science and Technology
Publication details, including instructions for authors and subscription information:
http://www.informaworld.com/smpp/title~content=t713708471

A Robust and Efficient Calculation Procedure for Determining Concentration Distribution of Multicomponent Mixtures

Shi Zeng[a]; Chuntong Ying[a]
[a] DEPARTMENT OF ENGINEERING PHYSICS, TSINGHUA UNIVERSITY, BEIJING, PEOPLES REPUBLIC OF CHINA

Online publication date: 20 March 2000

To cite this Article Zeng, Shi and Ying, Chuntong(2000) 'A Robust and Efficient Calculation Procedure for Determining Concentration Distribution of Multicomponent Mixtures', Separation Science and Technology, 35: 4, 613 — 622

To link to this Article: DOI: 10.1081/SS-100100179

URL: http://dx.doi.org/10.1081/SS-100100179

SEPARATION SCIENCE AND TECHNOLOGY, **35**(4), pp. 613–622, 2000

A Robust and Efficient Calculation Procedure for Determining Concentration Distribution of Multicomponent Mixtures

SHI ZENG* and CHUNTONG YING
DEPARTMENT OF ENGINEERING PHYSICS
TSINGHUA UNIVERSITY
BEIJING 100084, PEOPLE'S REPUBLIC OF CHINA

ABSTRACT

Determining the concentration distribution of mixture components in a separation cascade plays an important role in analyzing and optimizing separation performance. A new iterative method, called the "*q*-iteration method," is presented to solve the nonlinear difference equations describing the concentration distribution of a multicomponent mixture in separation cascades at steady state. Iteration takes place over quantity *q*, the ratio of the head concentration to tail concentration of certain components, other than the concentration itself in commonly used methods. Numerical tests are carried out for both short and long square cascades. Comparisons are made with the transient approach. The results show that the new method is both robust and efficient.

Key Words. Iterative method; Separation cascade; Multicomponent isotopic mixture

INTRODUCTION

It is desirable to have a robust and efficient method for determining the distribution of mixture components in a separation cascade in order to analyze and optimize its separation performance. The governing difference equations describing the distribution of mixture components at steady state are nonlinear, and therefore iterative methods are commonly used to solve the equations.

* To whom correspondence should be addressed.

Appropriate specification of initial values is important to achieve convergence. However, it is usually difficult to give appropriate initial values, especially in the case of complex cascades. So very often the method used either converges slowly or even fails. To overcome this problem, a new method, called "transient approach," has recently been reported. It solves a set of nonlinear partial differential-difference equations. For a commonly used iteration approach and the transient approach, see Ref. 1.

The nonlinear partial differential-difference equations employed in the transient approach can actually be used to describe the state change of cascades with time. Realizing that a cascade will eventually reach steady state from whatever its initial state, solving the time-dependent equations can lead us to the steady state. It still works when the iteration approach fails. However, the complexity of solution is increased compared with the iteration approach, and more computer time is required. It is desirable to use the transient approach if an understanding of the transient process is wanted. But in practice, it seems that the steady state is more interesting than the transient process. Furthermore, in an optimization process, the parameters, for example, the cut and the stage flow rate, may be chosen as optimization variables and their initial values, which are generated automatically by the optimization algorithm used, may be impractical. Consequently, the iteration approach cannot be used since it is easy to break down. The transient approach is not a good choice either, since impractical initial values result in a significantly greater consumption of computer time.

In this paper a new iterative method is presented for the solution of the steady-state nonlinear difference equations. It is simple as well as robust and efficient. Its performance is demonstrated numerically by comparing it with the transient approach in Ref. 1. We refer to this new method as the q-iteration method.

THE STEADY-STATE DIFFERENCE EQUATIONS

We consider the separation process of a multicomponent mixture in a gas centrifuge cascade, as illustrated in Fig. 1. Let the mixture contain N_c components. For simplicity, we assume that there is only one feed point and two withdrawals (product and waste). The feed rate is F, and the product and the waste rates are P and W, respectively. The concentrations of components in the feed are $C_{i,F}$, $i = 1, 2, \ldots, N_c$. For a cascade with N stages, according to the mass conservation for each component and the total mass, we easily obtain

$$\begin{aligned} L_n C'_{i,n} - (L_n + W)C''_{i,n+1} &= -WC''_{i,1}; \qquad 1 \le n < N_F \\ L_n C'_{i,n} - (L_n - P)C''_{i,n+1} &= PC'_{i,N}; \qquad N_F \le n < N \end{aligned} \tag{1}$$

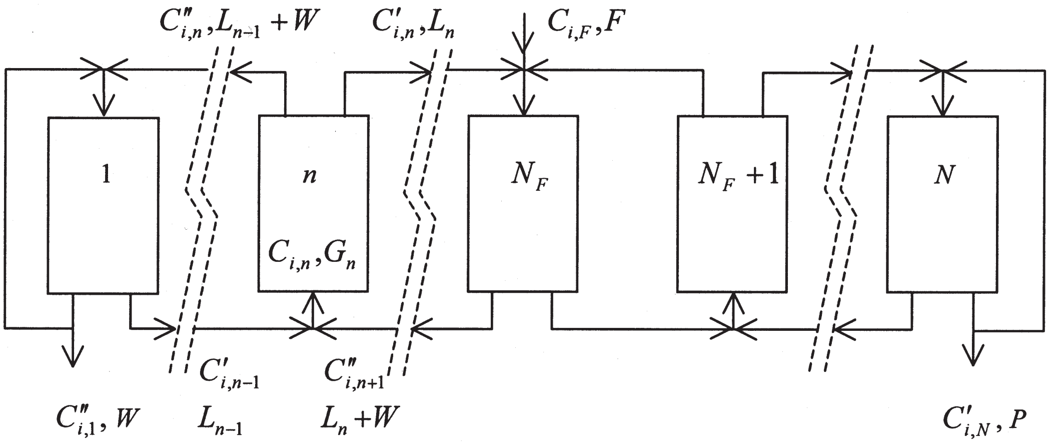

FIG. 1 A cascade with one feed and two withdrawals.

$$\begin{aligned} L_nC'_{i,n} + (L_{n-1} + W)C''_{i,n} &= (L_n + L_{n-1} + W)C_{i,n}; \qquad 1 \le n < N_F \\ L_nC'_{i,n} + (L_{n-1} + W)C''_{i,n} &= (L_n + L_{n-1} - P + F)C_{i,n}; \qquad n = N_F \\ L_nC'_{i,n} + (L_{n-1} - P)C''_{i,n} &= (L_n + L_{n-1} - P)C_{i,n}; \qquad N_F < n \le N \end{aligned} \tag{2}$$

$$PC'_{i,N} + WC''_{i,1} = FC_{i,F} \tag{3}$$

$$P + W = F \tag{4}$$

Here, $C_{i,n}$, $C'_{i,n}$, and $C''_{i,n}$ are the concentrations of components of the stage inflow, the head flow, and the tail flow at the nth stage, respectively; L_n is called the flow rate of the upflowing stream at the nth stage. The following well-known relationship (2) holds between the concentrations of any two components:

$$\gamma_{ij} = \frac{C'_i}{C''_i} / \frac{C'_j}{C''_j} = \gamma_0^{M_j - M_i} \tag{5}$$

γ_{ij} is called the separation factor, and γ_0 is known either through experiment or by separation analysis for a single stage. M_i and M_j are the molar weights of the ith and the jth components, respectively. The concentrations should satisfy the following conditions:

$$\sum_i C_{i,n} = 1, \qquad \sum_i C'_{i,n} = 1, \qquad \sum_i C''_{i,n} = 1 \tag{6}$$

Equations (1)–(5) determine the concentration distribution in a separation cascade and are the equations to be solved.

THE *q*-ITERATION METHOD

For simplicity of illustration and for the purpose of comparison with Ref. 1, we use a square cascade, i.e., the interstage flow rate G_n is a constant and is given. In reality, G_n may be chosen by means of optimization on separative power, or from other practical considerations. Most iterative methods use the concentration as the iteration variable, for instance, $C''_{i,n}$, as described in Ref. 1. But here we choose the ratio $C'_{k,n}/C''_{k,n}$ of the kth component as the iteration variable. Since the ratio is denoted by the symbol "q," it is named the q-iteration method. Noticing that the nonlinearality only appears in Eq. (5), this choice gives us a set of linear equations at an iteration step. With the initial values of $q_{k,n} = C'_{k,n}/C''_{k,n}$ given for each stage, we have from Eq. (5)

$$C'_{i,n} = q_{i,n} C''_{i,n} \tag{7}$$

with $q_{i,n} = q_{k,n}\gamma_0^{M_k - M_i}$. Substituting Eq. (7) into Eqs. (1) and (3), we get

$$\begin{aligned} WC''_{i,1} + L_n q_{i,n} C''_{i,n} - (L_n + W)C''_{i,n+1} &= 0; \qquad 1 \le n < N_F \\ L_n q_{i,n}\, C''_{i,n} - (L_n - P)C''_{i,n+1} - Pq_{i,N} C''_{i,N} &= 0; \qquad N_F \le n < N \\ WC''_{i,1} + Pq_{i,N}\, C''_{i,N} &= FC_{i,F}; \qquad n = N \end{aligned} \tag{8}$$

Using the third equation in Eq. (8) to eliminate $C''_{i,1}$ in the first equation yields

$$\begin{aligned} -(L_1 + W)C''_{i,2} - (W + L_1 q_{i,1})\frac{P}{W} q_{i,N}\, C''_{i,N} &= \\ -(W + L_1 q_{i,1})\frac{F}{W} C_{i,F}; &\qquad n = 1 \\ L_n q_{i,n}\, C''_{i,n} - (L_n + W)C''_{i,n+1} & \\ - Pq_{i,N}\, C''_{i,N} = -FC_{i,F}; &\qquad 2 \le n < N_F \\ L_n q_{i,n}\, C''_{i,n} - (L_n - P)C''_{i,n+1} - Pq_{i,N}\, C''_{i,N} = 0; &\qquad N_F \le n < N \end{aligned} \tag{9}$$

If we denote the above equation system as $\mathbf{Ax} = \mathbf{b}$, with $\mathbf{A}$ being the coefficient matrix, $\mathbf{x} = (C''_{i,2}, C''_{i,3}, \ldots, C''_{i,N})$, and $\mathbf{b}$ the right-hand side, matrix $\mathbf{A}$ has the following nonzero pattern:

$$\begin{array}{r|cccccc|} i = & 2 & 3 & 4 & \cdots & \cdots & N \\ n = 1 & * & & & & & * \\ 2 & * & * & & & & * \\ 3 & & * & * & & & * \\ \vdots & & & \ddots & \ddots & & \vdots \\ N-2 & & & & * & * & * \\ N-1 & & & & & * & * \end{array}$$

The two nonzero diagonals are denoted by α and β, respectively, and the $(N - 1)$th column $(i = N)$ by γ. Fairly accurate solution of this equation system plays an important role in making the method converge and is carried out as follows. We distinguish three cases and solve them separately, based on the consideration of solution accuracy as well as computational efficiency.

Case 1: $|\alpha_n| \leq |\beta_n|$ for All n

In this case, matrix **A** is decomposed into four submatrices, as shown below.

$$\begin{array}{r|cccccc|}
i = & 2 & 3 & 4 & \cdots & \cdots & N \\
\hline
n = 1 & * & & & & & * \\
2 & * & * & & & & * \\
3 & & * & * & & & * \\
\vdots & & & \ddots & \ddots & & \vdots \\
N - 2 & & & & * & * & * \\
N - 1 & & & & & * & *
\end{array} = \begin{pmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{pmatrix}$$

The unknown vector **x** and the right-hand side are decomposed accordingly into $\mathbf{x}_1 = (C''_{2,n}, C''_{3,n}, \ldots, C''_{i,N-1})$ and $\mathbf{x}_2 = (C''_{i,N})$, and $\mathbf{b}_1 = (b_1, b_2, \ldots, b_{N-2})$ and $\mathbf{b}_2 = (b_{N-1})$. So the equation system can be written as

$$\begin{pmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{pmatrix} \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{pmatrix} = \begin{pmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{pmatrix} \tag{10}$$

Therefore, the solution is easily obtained by

$$\begin{aligned} \mathbf{x}_2 = {} & (\mathbf{A}_{22} - \mathbf{A}_{21}\mathbf{A}_{11}^{-1}\mathbf{A}_{12})^{-1} \\ & \times (\mathbf{b}_2 - \mathbf{A}_{21}\mathbf{A}_{11}^{-1}\mathbf{b}_1), \qquad \mathbf{x}_1 = \mathbf{A}_{11}^{-1}(\mathbf{b}_1 - \mathbf{A}_{12}\mathbf{x}_2) \end{aligned} \tag{11}$$

Note that $\mathbf{A}_{22} - \mathbf{A}_{21}\mathbf{A}_{11}^{-1}\mathbf{A}_{12}$ is just a scalar, and the calculation of $\mathbf{A}_{11}^{-1}\mathbf{A}_{12}$ and $\mathbf{A}_{11}^{-1}\mathbf{b}_1$ uses forward substitution.

Case 2: $|\alpha_n| > |\beta_n|$ for All n

Moving the first row of matrix **A** to the last row, the resulting matrix is as follows:

$$\begin{array}{r|cccccc|}
i = & 2 & 3 & 4 & \cdots & \cdots & N \\
\hline
n = 1 & * & * & & & & * \\
2 & & * & * & & & * \\
3 & & & * & \ddots & & * \\
\vdots & & & & \ddots & * & \vdots \\
N - 2 & & & & & * & * \\
N - 1 & * & & & & & *
\end{array}$$

So diagonal α becomes the main diagonal. This equation system can be solved in the same way as given by Eqs. (11).

Case 3: For Cases Not Belonging to Either Case 1 or Case 2

In this case, $|\alpha_n| > |\beta_n|$ for some values of n, and $|\alpha_n| \leq |\beta_n|$ for other values of n. Handling this case in the same way as for Case 1 or Case 2 may deteriorate convergence or even cause divergence. Therefore, it has to be treated separately.

Since the matrix is sparse for computational efficiency, it is desirable to maintain its sparsity during solution. A symmetric Markowitz strategy (3) is used to choose appropriate pivots to achieve numerical stability and reduce fill-ins. The basic idea of the Markowitz strategy (4) is that during LU factorization of matrix **A**, element a_{ij} is chosen in the remaining unfactorized submatrix to be the pivot at a factorization step if the product $N_i N_j$ is minimum. Here, N_i and N_j are the numbers of nonzeros in the ith row and the jth column, respectively, of the submatrix after the factorization step. The readers are referred to Refs. 3, 4, and 5 for more details.

With $C''_{i,n}$ obtained, it is straightforward to calculate $C'_{i,n}$ from Eq. (7), and $C_{i,n}$ from Eq. (2). However, the solution most likely does not satisfy Condition (6) because $q_{k,n}$ is unlikely to be the correct value. In order to allow Condition (6) to be satisfied, $q_{k,n}$ needs to be modified. Let us consider Eq. (2). Summing up over i and taking into account Eq. (7) gives

$$\theta_n \sum_i q_{k,n} \gamma_0^{M_k - M_i} C''_{i,n} + (1 - \theta_n) \sum_i C''_{i,n} = \sum_i C_{i,n} \tag{12}$$

with θ_n given by

$$\theta_n = \begin{cases} L_n/(L_n + L_{n-1} + W); & 1 \leq n < N_F \\ L_n/(L_n + L_{n-1} - P + F); & n = N_F \\ L_n/(L_n + L_{n-1} - P); & N_F < n \leq N \end{cases} \tag{13}$$

We require

$$\sum_i q_{k,n} \gamma_0^{M_k - M_t} C''_{i,n} = \sum_i C_{i,n} \tag{14}$$

namely $\sum_i C'_{i,n} = \sum_i C_{i,n}$. On convergence, this implies $\sum_i C''_{i,n} = \sum_i C_{i,n}$ because of Eq. (12). Therefore, from Eqs. (8) we can conclude Condition (6) is satisfied. Equation (14) gives

$$q_{k,n} = \frac{\sum_i C_{i,n}}{\sum_i \gamma_0^{M_k - M_i} C''_{i,n}} \tag{15}$$

The iteration carries on to the next step using the new value of $q_{k,n}$ given by Eq. (15).

Weighting is sometimes necessary for difficult cases such as very long cascades and small P (or W), and is carried out by

$$q_{k,n} = (1 - \omega)q_{k,n}^{\text{old}} + \omega q_{k,n}^{\text{new}} \tag{16}$$

with $0 < \omega \leq 1$ the weighting factor, $q_{k,n}^{\text{old}}$ the value of $q_{k,n}$ at the previous iteration step, and $q_{k,n}^{\text{new}}$ given by Eq. (15). The termination criterion is to satisfy

$$\max_n(|\sum_i C_{i,n} - 1|, |\sum_i C'_{i,n} - 1|, |\sum_i C''_{i,n} - 1 |) \leq \varepsilon \tag{17}$$

where ε is a small given number.

EXAMPLES AND COMPARISON

To test the method, we choose the following two problems. Problem 1 is a square cascade with the constant stage inflow rate $G_n/F = 10$ and $P/F = 0.2$. The number of stages is $N = 20$ and the feed position is $N_F = 2$. Problem 2 is also a square cascade, but this time with $G_n/F = 10$, $P/F = 0.85$, $N = 100$, and $N_F = 10$. The choice of such a strange cascade is only for the purpose of test and comparison and is of little practical significance. In order to have sufficient complexity for the test, we choose Xe as the process gas because it has a big number of isotopes ($N_c = 9$) and its separation factor is quite large ($\gamma_0 = 1.4$). For the two problems, the cut of the first stage is determined by $\theta_1 = 0.5(1 - W/G_1)$. So the flow rates of upflowing streams are given from the mass conservation:

$$L_0 = (1 - \theta_1)G_1 - W$$
$$L_n = G_n - L_{n-1} - W; \qquad 1 < n \leq N_F$$
$$L_n = G_n - L_{n-1} + P; \qquad N_F < n \leq N$$

All calculations are carried out on a PC with a Pentium Pro-200 processor. The concentration distribution for Problem 1 is given in Fig. 2. Figure 3 presents the convergence history of the q-iteration method and the transient approach. The q-iteration method uses the left and the bottom coordinates (indicated by Q-iteration), and the transient approach uses the top and the right coordinates (indicated by T-approach). The error for the q-iteration method is defined as

$$err = \max_n(|\sum_i C_{i,n} - 1|, |\sum_i C'_{i,n} - 1|, |\sum_i C''_{i,n} - 1|)$$

and that for the transient approach as

$$err = |\sum_i (C'_{i,n}P + C''_{i,n}W - C_{i,F}F)|$$

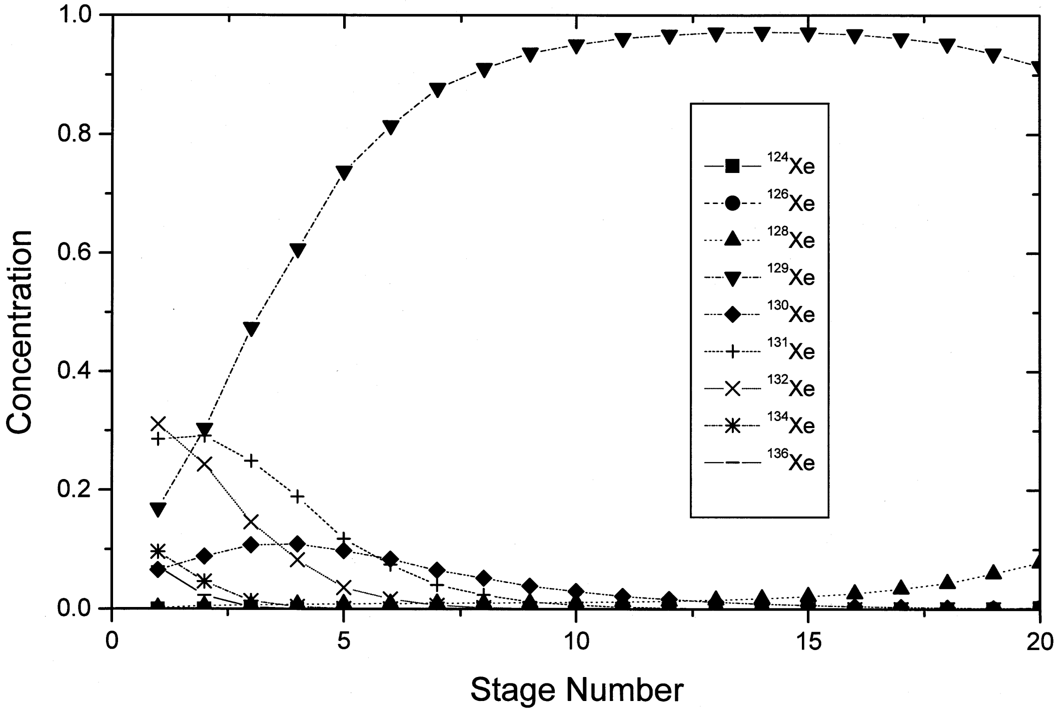

FIG. 2 Concentration distribution of Xe isotopes in a 20-stage cascade.

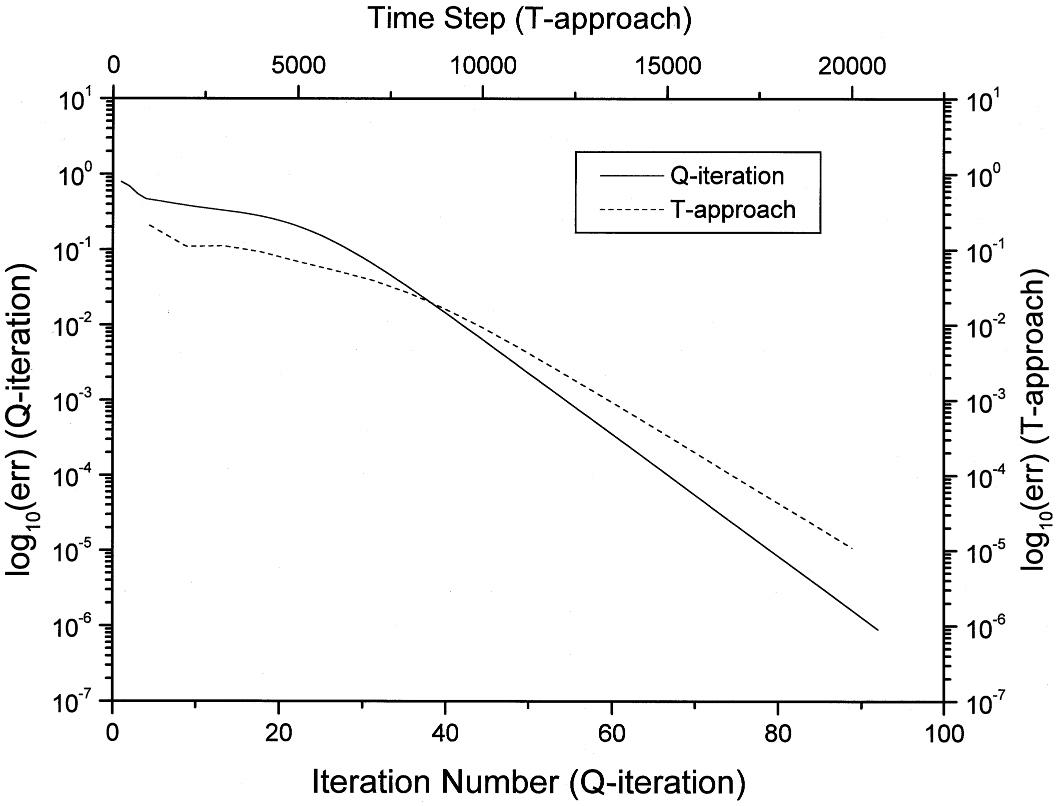

FIG. 3 Convergence history of the *q*-iteration method and the transient approach for a 20-stage cascade.

In order to have a fair comparison, both methods are required to give six significant digits for the concentration. Furthermore, the two methods should be made to work at their best performance. For the q-iteration method, a larger ω gives faster convergence; for the transient approach, a larger Δt uses a smaller number of time steps. But there may be cases in which the methods fail for too large ω and too large Δt. So the weighting factor ω for the q-iteration method and the time step Δt for the transient approach should be chosen as large as possible to achieve the best performance. Based on this consideration, here ω is taken to be 1 and Δt to be 0.006457. The q-iteration method uses 92 iterations and takes 0.22 seconds whereas the transient approach needs 20,120 time steps and takes 23 seconds. So the q-iteration method is about 100 times faster than the transient approach. The convergence history is shown in Fig. 4 for Problem 2. Now the largest possible values for ω and Δt are about 0.3 and 0.007972, respectively. For the q-iteration method, 452 iterations are performed, which takes 3.24 seconds. But for the transient approach, 57,862 time steps are carried out, which takes 329 seconds. Again, there is a factor of about 100 times difference in computing time.

Generally speaking, convergence can be achieved by adjusting ω for the q-iteration method and Δt for the transient approach. The larger the value of ω

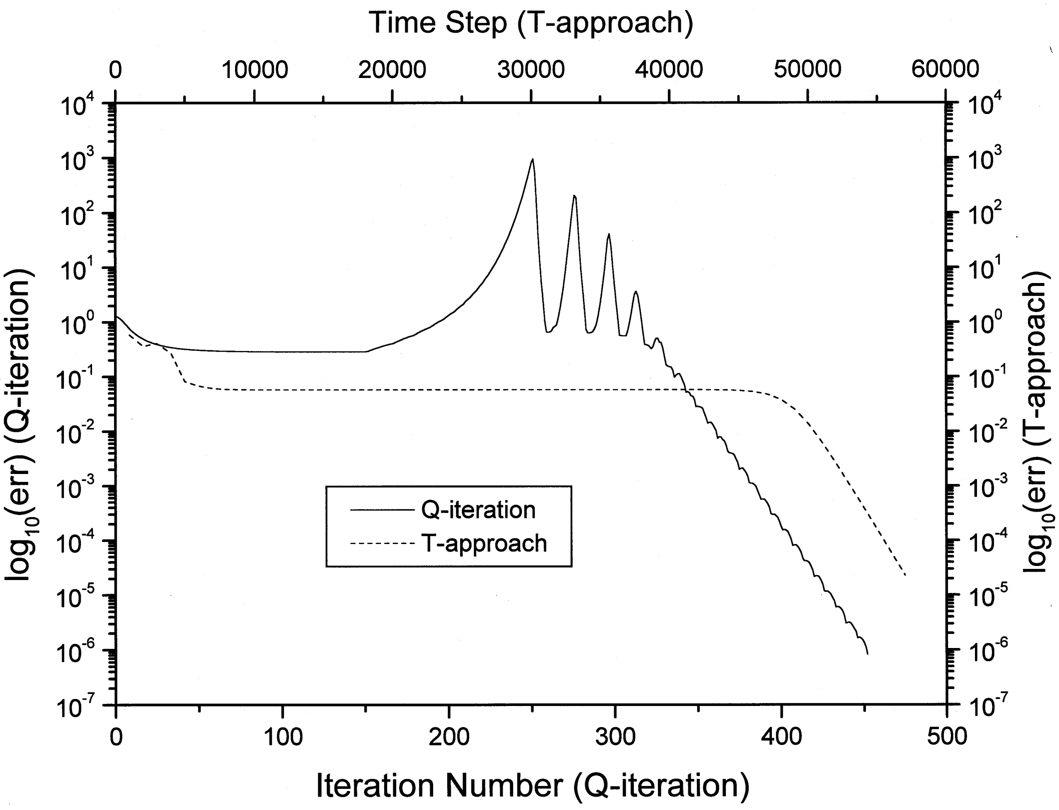

FIG. 4 Convergence history of the q-iteration method and the transient approach for a 100-stage cascade.

or Δt is, the less CPU time is spent. But in order to avoid the trouble of choosing different values of ω for different cases, ω can be fixed at a smaller value, say 0.3. Although this may slow down convergence for cases where a larger ω can be used, the *q*-iteration method still works very efficiently. For example, with $\omega = 0.3$ instead of 1, solution of Problem 1 uses 355 iterations and takes 0.44 seconds.

CONCLUSIONS

A new iteration method, called the *q*-iteration method, has been designed to determine the concentration distribution of a multicomponent mixture in separation cascades. Unlike commonly used iteration methods, which iterate on the concentration, iteration in the *q*-iteration method is carried out on the ratio of the head concentration to the tail concentration at each stage of a certain component, which can be chosen arbitrarily. For given values of the ratios, the resulting difference equations are easily solved. With the newly obtained concentration distribution, the ratios are corrected, based on the requirement that the summation of concentrations for all components at each stage is equal to unity and the mass conservation. Weighting is necessary in some cases to obtain convergence, but the weighting factor may be fixed at a certain value to suit all cases. Two test problems are used to demonstrate that the *q*-iteration method is both robust and efficient.

ACKNOWLEDGMENTS

This research is supported by the National Natural Science Foundation of the People’s Republic of China (No. 59676020). The authors thank Dr. H. J. Wu for her code for the transient approach.

REFERENCES

1. H. J. Wu, C. T. Ying, and G. J. Liu, "Calculation Methods for Determining the Distribution of Components in a Separation Cascade for Multicomponent Mixture," *Sep. Sci. Technol.*, *33*(6), 887–898 (1998).
2. C. Ying, Z. Guo, and H. G. Wood, "Solution of the Diffusion Equations in a Gas Centrifuge for Separation of Multicomponent Mixtures," *Ibid.*, *33*(18), 2455–2471 (1996).
3. K. Crowe, Y. A. Fan, J. Li, D. Neaderhouser, and P. Smith, "A Direct Sparse Linear Equation Solver Using Linked List Storage," *IMSL Technical Report 9006*, IMSL, Houston, TX, 1990.
4. H. M. Markowitz, "The Elemination Form of the Inverse and Its Application to Linear Programming," *Manage. Sci.*, *3*, 255–269 (1957).
5. I. S. Duff, A. M. Erisman, and J. K. Reid, *Direct Method for Sparse Matrices*, Clarendon Press, Oxford, 1986.

Received by editor January 22, 1999
Revision received July 1999