ACCELERATION ALGORITHMS FOR PROCESS DESIGN SIMULATIONS

by

ANOUSHTAKIN ARMAN

Bachelor of Science in

Chemical Engineering

Oklahoma State University

Stillwater, Oklahoma

May, 1985

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
MASTER OF SCIENCE
December, 1986

ACCELERATION ALGORITHMS FOR PROCESS DESIGN SIMULATIONS

Thesis Approved:

Thesis Adviser

Dean of the Graduate College

ABSTRACT

Direct Substitution Methods for convergence in simulation software are often slow and time consuming. Convergence can be speeded up using an acceleration algorithm. Three acceleration algorithms were tested on MAXISIM, a chemical process design simulation package developed at Oklahoma State University. The algorithms tested were Wegstein's Method, Dominant Eigenvalue Method (DEM), and the General Dominant Eigenvalue Method (GDEM). Eight process models were tested ranging from non-oscillatory to very oscillatory systems using a variety of combinations of chemical process units at different conditions.

The best result was found using GDEM, ranging from no improvement for the very oscillatory systems to over 90 % reduction in the number of iterations in the case of a non-oscillatory system. An average saving of 45 % in cpu time can be achieved for a typical process model.

ACKNOWLEDGEMENTS

I wish to express my sincere gratitude to all the people who helped me in this work at Oklahoma State University. In particular, I am indebted to my principal adviser, Dr. Ruth C. Erbar for her guidance, encouragement, and moral support to continue on the path of education. I am also thankful to Dr. J. Wagner and Dr. M. Seapan for taking on the task and the responsibility as my committee members.

The presence and the help of Danny Friedemann and Carlos Ruiz on computers was especially appreciated. Also my deepest appreciation goes to the School of Chemical Engineering at Oklahoma State University for the financial support I received during the course of this work.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

A - Linearized approximation to F

E - Accuracy desired

F(X) - Feed stream

FR - Calculated feed of a broken recycle stream

H - The negative inverse of the Jacobian matrix

I - The identity matrix

PR - Calculated product of a broken recycle stream

T - The transpose matrix

W - Weighting matrix

X - Feed stream to a process unit

e - An arbitrary point near n

n - Number of iterations performed

o - The initial guess

s - The absolute solution

v - Number of coefficients to estimate $\hat{\mu}_j$

α - Damping factor in DEM

λ - Eigenvalue of A

Δ - Forward difference operator

$\hat{\mu}_j$ - jth eigencoefficient of A

$\hat{}$ - Estimated value

 - Temperature, F

 - Pressure, psia

 - Stream number

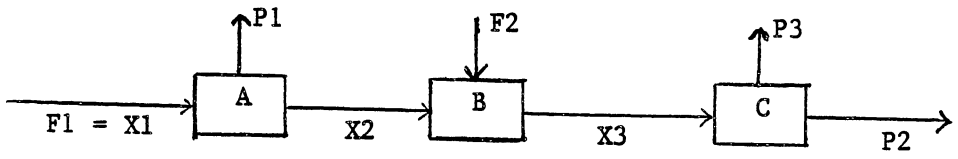
CHAPTER I

INTRODUCTION

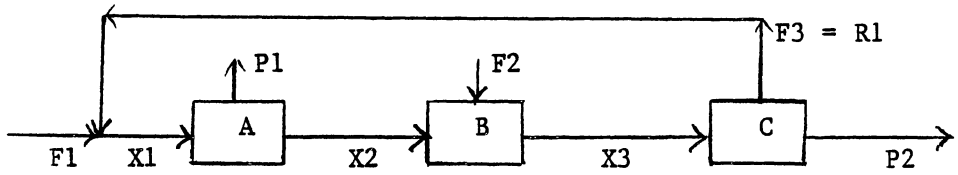
A steady state process design simulation is a mathematical model representing a process. The independent variables or the specified conditions are identified and the dependent variables are calculated.

In a process simulation with no recycle streams the calculation is usually straight forward. The calculations on each process unit are done individually and sequentially and the process is completed in one iteration (Figure 1(A)). However a process with one or more recycle streams necessitates the use of an iterative procedure for convergence. The convergence criterion is usually a specified tolerance in the change of properties and/or rates between two sequential iterations on the recycle stream (9). Most process simulators use the sequential modular architecture to establish a logical method for solving for the unknown variables in the system.

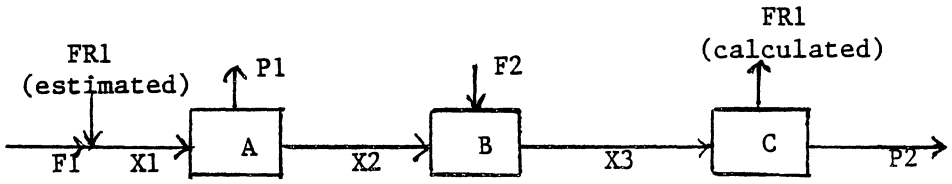
Sequential modular architecture is a concept where the recycle streams are conceptually broken and treated as products, PR, of the originating unit and feeds, FR, to the destination unit. The calculations are performed sequentially as if no recycle stream exists. This procedure is repeated



(A) Process With no Recycle



(B) Process Containing a Recycle Stream



(C) Process Containing a Recycle Stream
Conceptually Broken into Feeds
and Products

Fn - External Feed Streams
Pn - External Product Streams
Xn - Internal Product/Feed Streams
FRn - Recycle Streams

Figure 1. Sequential Modular Architecture

until the feed and the product agree within a set tolerance (Figures 1(B and C)).

In many instances the process of convergence to the set tolerance becomes a very time consuming one involving millions of calculations and thus the incentive to use an acceleration algorithm to accelerate the convergence or reduce the number of calculations becomes great. The variables for the acceleration algorithm can be the individual mass flow rates of the components in any stream in the process, the temperature, pressure, quality or any other property of the stream that is changing with every iteration. These variables almost always have a non-linear dependent relationship with respect to each other that can be likened to a set of non-linear dependent equations. An example is the relationship of the feed stream of the process simulation to the product streams.

The relationship between the feed, X, and a product stream, F(X), can be mathematically represented as

$$F(X) = X \tag{1-1}$$

In a steady state process simulation the function, F, is usually too complex to be expressed mathematically. The problem is to find Xs such that

$$F(X) = Xs \tag{1-2}$$

where Xs is the solution matrix for which the process will converge absolutely. Although it may be impossible to prove mathematically that a solution exists it will be assumed that it exists and that it is unique unless proven otherwise. Often a process does not reach a solution due to preset conditions imposed on the process that are not realistic, i.e., physically impossible. However in practice one may only wish to approximately determine the solution in order to save computer time so that

$$Xn \approx Xs \tag{1-3}$$

and

$$\sum (Xn - X(n-1))^2 < E \tag{1-4}$$

where, E, is the tolerance desired.

The objective is to find an algorithm that can be used to accelerate the convergence of Xn to Xs and to successfully employ that algorithm in a chemical process simulation software.

CHAPTER II

LITERATURE SURVEY

NUMERICAL METHODS

In recent years there has been a great deal of research in the area of acceleration algorithms. However, only a few of these methods will be discussed below. The advantage of having an automatic means of accelerating the solution in a computer simulation must be obvious to the reader. A few specific questions that should be kept in mind when discussing numerical methods are 1) When will the acceleration be applied ? 2) Is the algorithm stable ? and 3) How much computer time can be saved ?

Direct Substitution (D.S.)

In direct substitution the previous value of X is substituted in the function vector

$$X(n+1) = F(Xn) \quad . \qquad (2\text{-}1)$$

This method is not really an acceleration algorithm at all and is often very slow to converge. However, direct substitution is very stable especially where oscillatory behavior exists in the system.

Wegstein's Method

Wegstein's method for multivariable programs is a secant method approximation first proposed by Aitkin (1) where the new estimate for Xn is estimated as follows

$$X(n+1) = Xn - F(Xn)(Xn-X(n-1)) \;/\; (F(Xn-F(Xn-1))) \quad (2\text{-}2)$$

Aitkin's method was later modified by Wegstein (14) and Kliesh (9) until Graves (8) proposed the following equivalent expression for X(n+1) where the function, F, has been linearized (Figure 2).

$$X(n+1) = (1-q)\, F(Xn) + q\, Xn \quad (2\text{-}3)$$

where

$$q = s \;/\; (s-1) \quad (2\text{-}4)$$

and

$$s = (F(Xn) - FX(n-1)) \;/\; (Xn - X(n-1)) \quad . \quad (2\text{-}5)$$

The advantage of the Graves expression is that a limit can be set on the parameter q . Note that if Xn = X(n+1) or if s = 1 the calculation of X(n+1) becomes impossible. For various values of q the characteristics of the Wegstein are

q = 0	successive substitution
q < 0	can speed convergence but also introduces instability
q > 0	slow, stable convergence.

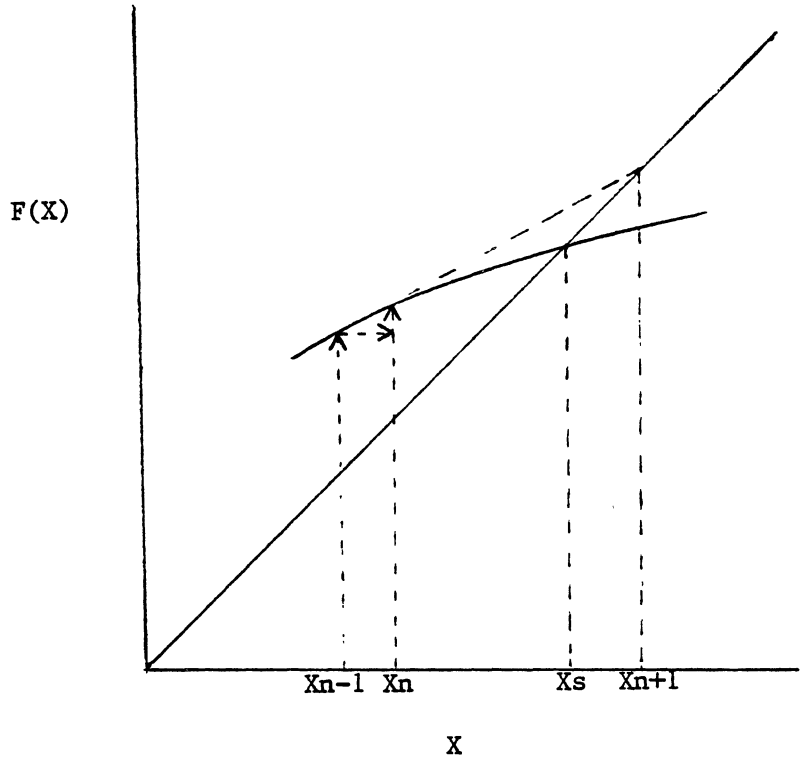


Figure 2. Graphical Illustration of the Wegstein's Method

All accelerating algorithms assume a linearity of the matrix near the solution. Although this may be a good assumption in most cases there are exceptions to this rule (APPENDIX A). Note that Wegstein's method is applied to every variable in the matrix separately. Thus , ignoring the interaction between the variables is the biggest deficiency of the method. This characteristic of the method can lead to an oscillatory behavior that can result in the divergance of the solution.

The oscillatory behavior of the Wegstein can be partially corrected by setting an upper and a lower limit on the value of q .

$$q\ (\text{min.}) < q < q\ (\text{max.})$$

Called the bounded Wegstein, this method is similar to introducing a damping factor to counter the oscillatory behavior of the method and thus assure the convergence. Another detriment of the method is that there are no specific criteria to help determine when the acceleration should be used. These problems will be discussed in the next chapter.

Dominant Eigenvalue Method (DEM)

If the iteration is approximated by a first order Taylor

series expansion of equation 2-1 about an arbitrary point, Xe, in the neighborhood of Xn the linear matrix becomes

$$X(n+1) = A\,Xn + b \qquad (2\text{-}6)$$

where

$$A = \left(\partial F / \partial X \right), \quad @\ X = Xe \qquad (2\text{-}7)$$

and

$$b = F(Xe) - A\,Xe \ . \qquad (2\text{-}8)$$

Orbach and Crowe's DEM (11) is a convergence scheme based on the assumption that the largest eigenvalue in A dominates the solution. It is necessary at this point to introduce a few definitions. From equation 2-6 the function F(X) can be expressed as AXn where the eigenvalues of X are defined such that they satisfy the equality, $AXn = \lambda Xn$ where X is called the eigenvector, λ is called the eigenvalue of X, and all the eigenvalues of X are called the eigenrow of X. If the eigenvalues, λj, of A are labeled in descending order of the absolute magnitude, the only necessary and sufficient condition for convergence would be that

$$| \lambda 1 | < 1 \qquad (2\text{-}9)$$

where, $\lambda 1$, is the dominant eigenvalue (7).

The solution to equation 2-6 is in general

$$Xn - Xs = A^{n} \ (Xo - Xs) \qquad (2\text{-}10)$$

and in particular

$$Xn - Xs = \sum_{j=1}^{m} Cj\ Zj\ \lambda_j^n \qquad (2\text{-}11)$$

where Xo is the initial guess and Xs is X at the solution if all λ j are distinct. Here

$$Xs = (I - A)^{-1}\ b \qquad (2\text{-}12)$$

and

$$Cj = Wj^T\ (Xo - Xs) / (Wj^T\ Zj)\ , \qquad (2\text{-}13)$$

and Zj and Wj are the eigenvectors and eigenrows of λ j.

In a monotonic convergence near the solution equation 2-11 becomes approximately a geometric progression of the solution of the form

$$Xn - Xs = C1\ Z1\ \lambda_1^n\ . \qquad (2\text{-}14)$$

From equation 2-14 it can be shown that

$$\Delta\ Xn \equiv Xn - X(n-1) = C1\ Z1\ ((\lambda\ 1) - 1)\ \lambda_1^{n-1} \qquad (2\text{-}15)$$

and that the ratio of the two norms becomes

$$|\ \lambda_1\ | = \|\ \Delta\ Xn\ \| / \|\ \Delta\ X(n-1)\ \|\ . \qquad (2\text{-}16)$$

Combining equations 2-14 and 2-15 the apparent solution becomes

$$X(n+1) = X(n-1) + \alpha\ (Xn - X(n-1)) / (1 - \lambda\ 1) \qquad (2\text{-}17)$$

where α is the damping factor introduced to the equation to suppress oscillation. Note that if $\lambda 1$ is close to unity the correction becomes very large and convergence very slow. Also if $\lambda 1 < 0$ the correction falls between Xn and X(n-1).

The only necessary condition in DEM for convergence being $| \lambda 1 | < 1$. The stability of the method would then be directly proportional to the stability of $\lambda 1$. Orbach and Crowe (11) recommended the percentage change of λ as a measure of stability for the algorithm where

$$\Delta \lambda = (\lambda n - \lambda (n-1)) \, 100 \, / \, \lambda (n-1) \quad . \qquad (2\text{-}18)$$

Thus the criteria for acceleration become that $| \lambda 1 | < 1$ and that two successive eigenvalues differ by no less than a preset value, $\Delta \lambda$.

General Dominant Eigenvalue Method (GDEM)

Crowe and Nishio (5) proposed a more effective convergence promotion also based on the eigenvalues of the solution matrix. Starting with the basic linear form of equation 2-6 in terms of the forward differences

$$Xn = A \, X(n-1) \quad . \qquad (2\text{-}19)$$

The characteristic equation of A is

$$| \lambda I - A | = \sum_{j=0}^{m} \mu_j \lambda^{m-j} = 0 \qquad (2\text{-}20)$$

where m is the dimension of the matrix and μ_j is the eigencoefficient. Also,

$$\mu_j \equiv (-1)^j \sum \lambda_{i1} \lambda_{i2} \ldots \lambda_{ij} \qquad (2\text{-}21)$$

where

$$1 < j < m \ ,$$

$$1 < i1 < i2 < \ldots < ij < m \ ,$$

and

$$\mu_0 \equiv 1 \ .$$

From the Cayley-Hamilton theorem (10), A satisfies equation 2-20 so that

$$\sum_{j=0}^{m} \mu_j A^{m-j} \Delta X(n-m) = 0 \ . \qquad (2\text{-}22)$$

Repeated use of equations 2-19 and 2-22 gives

$$\sum_{j=0}^{m} \mu_j \Delta X(n-j) = 0 \ . \qquad (2\text{-}23)$$

If the eigenvalues are labeled in ascending order of magnitude and if we assume that only v of them were large enough to dominate the iteration, it then follows that

$$\sum_{j=v+1}^{m} \mu_j \Delta X(n-j) = 0 \ . \qquad (2\text{-}24)$$

The iteration is thus confined to a v dimensional subspace. An approximation to equations 2-22 and 2-23 gives

$$\sum_{j=0}^{v} \hat{\mu}_j \, \Delta X(i-j) = 0 \qquad (2\text{-}25)$$

where

$$i = n \, , \, (n+1) \, , \, \ldots$$

and $\hat{\mu}_j$ is an approximation to the real value of μ_j. Also μ_j is estimated by taking the derivative of the square norm with respect to μ_k and setting them equal to zero. Thus

$$\sum_{j=0}^{v} \hat{\mu}_j \; bjk = 0 \qquad (2\text{-}26)$$

where

$$k = 1 \, , 2 \, , \ldots \, , v$$

and

$$bij = < \Delta X(n-j) \, , \, \Delta X(n-k) > \quad . \qquad (2\text{-}27)$$

The inner product, $< X,Y >$, is defined by

$$< X,Y > = X^{T} \, Wy \qquad (2\text{-}28)$$

where W is the weighting matrix, usually the identity matrix. $\hat{X}s$ is the limit to convergence as Xn approaches infinity.

Thus equation 2-25 becomes

$$\hat{X}s - X(n+1) = \sum_{i=n+1}^{\infty} \Delta Xi = - \sum_{j=1}^{v} \hat{\mu}_j \sum_{i=n+1}^{\infty} \Delta X(i-j) \quad . (2\text{-}29)$$

Rearranging equation 1-29

$$\hat{X}s = \sum_{j=0}^{V} \mu j \; X(n+1-j) \; / \; \sum_{j=0}^{V} \hat{\mu} j \quad . \tag{2-30}$$

For V = 1, equation 2-30 reduces to

$$\hat{X}s = Xn + \Delta Xn \; / \; (1 + \hat{\mu} 1) \tag{2-31}$$

where

$$\hat{\mu} 1 = - b01 \; / \; b11 \quad . \tag{2-32}$$

Orbach and Crowe used $\hat{\lambda} 1 = (b00 \; / \; b11)^{1/2}$ from the Cauchy-Schwartz inequality ,

$$| \hat{\mu} 1 | < | \hat{\lambda} 1 | \tag{2-33}$$

for convergence. Thus with GDEM we have avoided the use of a damping factor and can use the full promotion step of the accelerator. A similar criterion for acceleration was used for GDEM as for DEM where, $\Delta \hat{\mu}$ is defined

$$\Delta \hat{\mu} = (\hat{\mu} n - \hat{\mu} (n-1)) \; 100 \; / \; \hat{\mu} (n-1) \quad . \tag{2-34}$$

Newton and Quasi-Newton Methods

The classical approach to solving non-linear simultaneous equations is the Newton method where the solution of the equations of the form

$$F(X) = 0 \tag{2-35}$$

is

$$X(n+1) = Xn - Jn^{-1} \; F(Xn) \tag{2-36}$$

where Ji is the Jacobian matrix of the first partial derivatives, (∂ F(X)/ ∂ X), evaluated at Xn (4). As was mentioned earlier the exact value of the Jacobian is almost never known in a process simulation. Therefore the Jacobian is usually approximated by the first difference of the matrix.

Quasi-Newton methods emerge as techniques to evaluate and update the Jacobian. In the Broyden's Method (2) the Jacobian is updated as follows;

$$H(n+1) = Hn - ((Hn\ Yn - Qn)\ Pn\ /\ Pn\ Yn\ ) \qquad (2\text{-}37)$$

where

$$Hn = -\ Jn^{-1} \qquad (2\text{-}38)$$

$$Pn = Hn\ Qn \qquad (2\text{-}39)$$

$$Qn = X(n+1) - Xn \qquad (2\text{-}40)$$

and

$$Yn = F(X(n+1)) - F(Xn)\ . \qquad (2\text{-}41)$$

Soliman (13) describes variations to equation 41 which are simpler, more efficient, and require less computer storage. These variations of Quasi-Newton methods can be divided into two categories. One in which the Jacobian is assumed to be the identity matrix and the other where the Jacobian is approximated by the first difference of the matrix. In the first category of the Quasi-Newton methods the improvement over the eigenvalue methods is not consider-

able. In the second category, although there is a considerable improvement made in convergence, there are however two disadvantages. 1) m iterations are necessary to determine the first approximation to the Jacobian. Thus for a fifty variable matrix fifty iterations will be required before a next guess could be made. 2) Considerable amount of computer time and storage will be necessary to store and invert the Jacobian. For these reasons Quasi-Newton methods will not be discussed as suitable candidates for the acceleration algorithm.

CHAPTER III

DISCUSSION AND RESULTS

To give the reader a sense of the relative strength of the acceleration algorithms a simple nonlinear classical problem called the Pipe Network was chosen. The algorithms tested were the Bounded Wegstein, DEM, and GDEM against Direct Substitution (DS). Process Model 1 or the pipe network consists of 5 horizontal pipes with 5 nodes (Figure 3). The pressure drop is given by the fanning equation (3)

$$Pi - Pj = Fm \rho Um^2 L / 2 D \qquad (3\text{-}1)$$

where, Fm is a dimensionless moody friction factor, ρ is the liquid density, Um is the mean velocity and L and D are the length and diameter of the pipe respectively. Given a flow rate, Q, where

$$Q = (\pi D^2 / 4) Um \quad . \qquad (3\text{-}2)$$

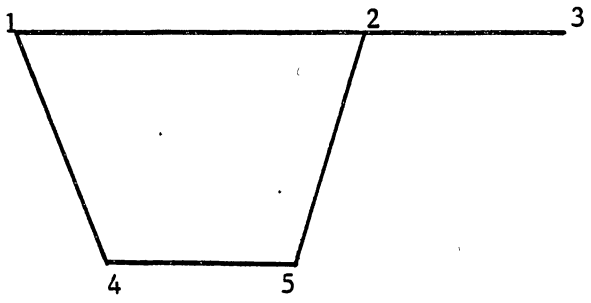
Equation 3-1 becomes

$$Pi - Pj = 8 Fm \rho Q^2 L / \pi^2 D^5 \qquad (3\text{-}3)$$

$$= C L Q^2 / D^5 \qquad (3\text{-}4)$$

where, C is a constant. Note Fm can be assumed constant in highly turbulent regions (i.e. low C values).

MODEL 1



C12 = C23 = 3.7326E-4

C14 = C45 = C52 = 5.905E-5

P1 = 50 , P3 = 0

Initial estimates of pressures are

P2 = 20 , P4 = 40 , P5 = 30

Figure 3. The Pipe Network

Let

$$Cij = C\ Lij\ /\ Dij^{5} \tag{3-5}$$

then

$$|\ Pi - Pj\ | = Cij\ Qij^{2} \tag{3-6}$$

where Qij is the flow rate between the nodes i and j. Equation 3-6 can be rearranged and since the sum of the flow rates is zero at any node,

$$Qij = (Pi - Pj)\ (1\ /\ (Cij\ |\ Pi-Pj\ |\ ))^{1/2} = 0\ . \tag{3-7}$$

Equation 3-7 can be rearranged to give

$$Pj = \sum Aij\ Pi\ /\ \sum Aij \tag{3-8}$$

where

$$Aij = (Cij\ |\ Pi - Pj\ |\ )^{1/2}\ . \tag{3-9}$$

The trial and error computation is performed as follows. An estimate of Aij is made using the previous values of the pressure at the nodes using equation 3-9. Then a new estimate of the pressure can be made by using equation 3-8.

$$Fi = Pj - \sum Aij\ Pi\ /\ \sum Aij \tag{3-10}$$

where Fi approaches zero as the solution converges. The error is estimated by the equation;

$$E = F2^{2} + F4^{2} + F5^{2}\ . \tag{3-11}$$

The results of each method is plotted in Figure 4. Wegstein's and DEM were optimized for best results (APPENDIX B and C). The results basically duplicate the findings of Soliman (13) where GDEM shows the best convergence of the problem.

The first step in finding the best acceleration algorithm is to find a typical chemical process model that can be representative of the type of models used in chemical engineering. All the chemical process models were tested using MAXISIM, a process design simulation package developed at Oklahoma State University (6). Process Model 2 is such a model, actually part of a real process system modified for our purposes (Figure 5). For Wegstein's method Q is damped between 0 and -5, for DEM. α = 0.9 and $\Delta\hat{\lambda}$ = 5%, and for GDEM $\Delta\hat{\mu}$ = 5%. The stream accelerated is # 3 where the error or the tolerance in the process simulations is defined as

$$E = | \Delta X | = \sum (X(i,n) - X(i,n-1))^2 \quad . \quad (3\text{-}12)$$

The results of the model 2 calculations are plotted in Figure 6. Again GDEM shows the best results. In fact the best results for GDEM were obtained using $\Delta\hat{\mu}$ = 1%. The percentage change in the eigencoefficient, $\Delta\hat{\mu}$, is in reality a measure of how accurate the next estimate will be. For example, a large $\Delta\hat{\mu}$ means the acceleration will be

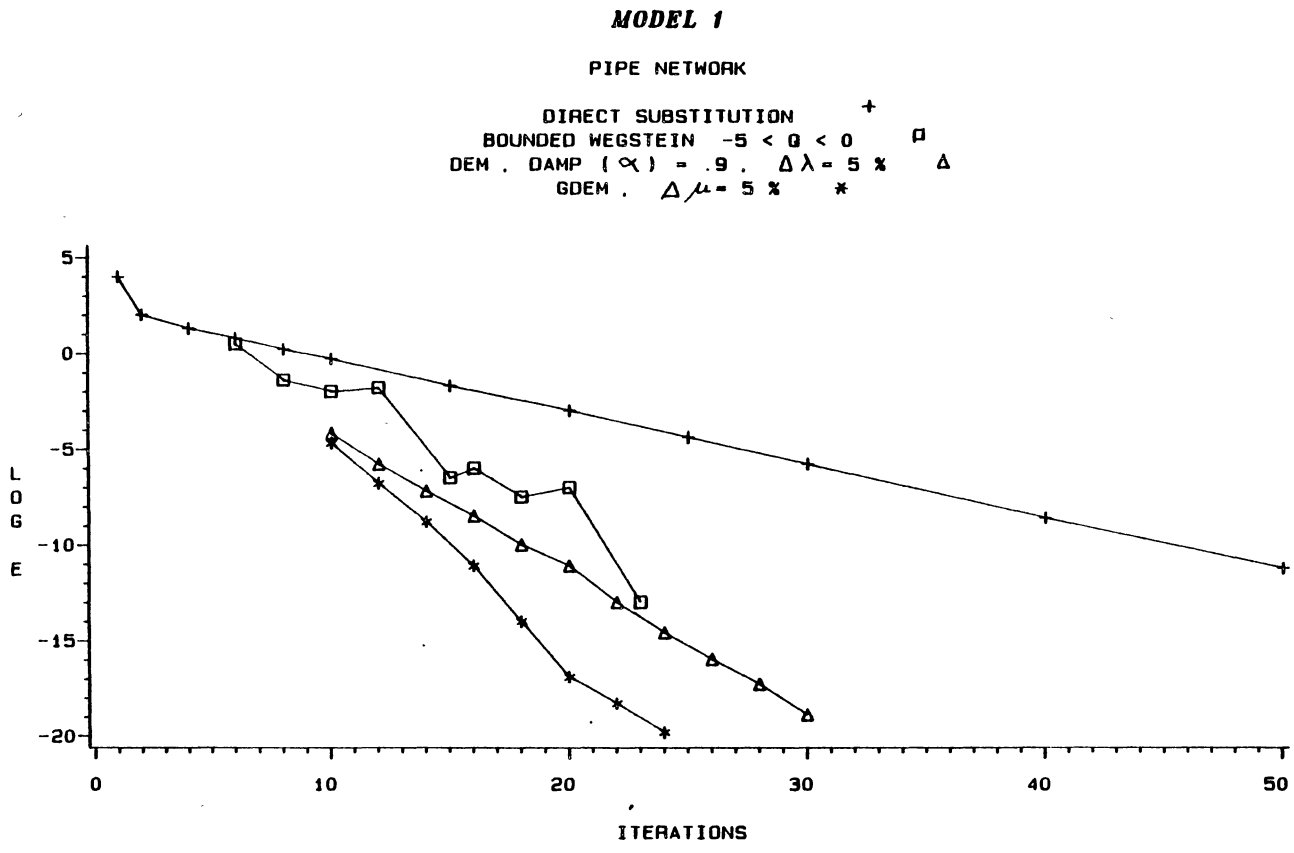


Figure 4. Comparison of the Three Methods in the Pipe Network

MODEL 2

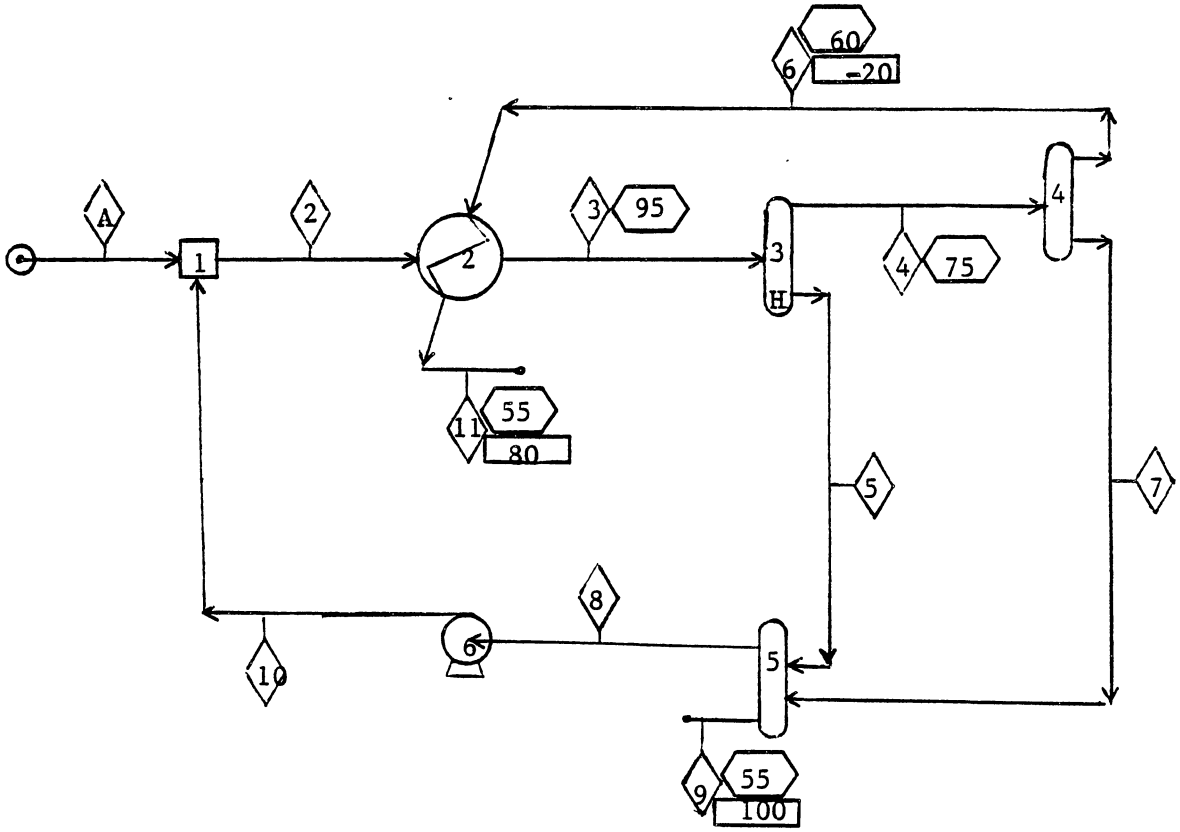


Figure 5. A Typical Chemical Process Model

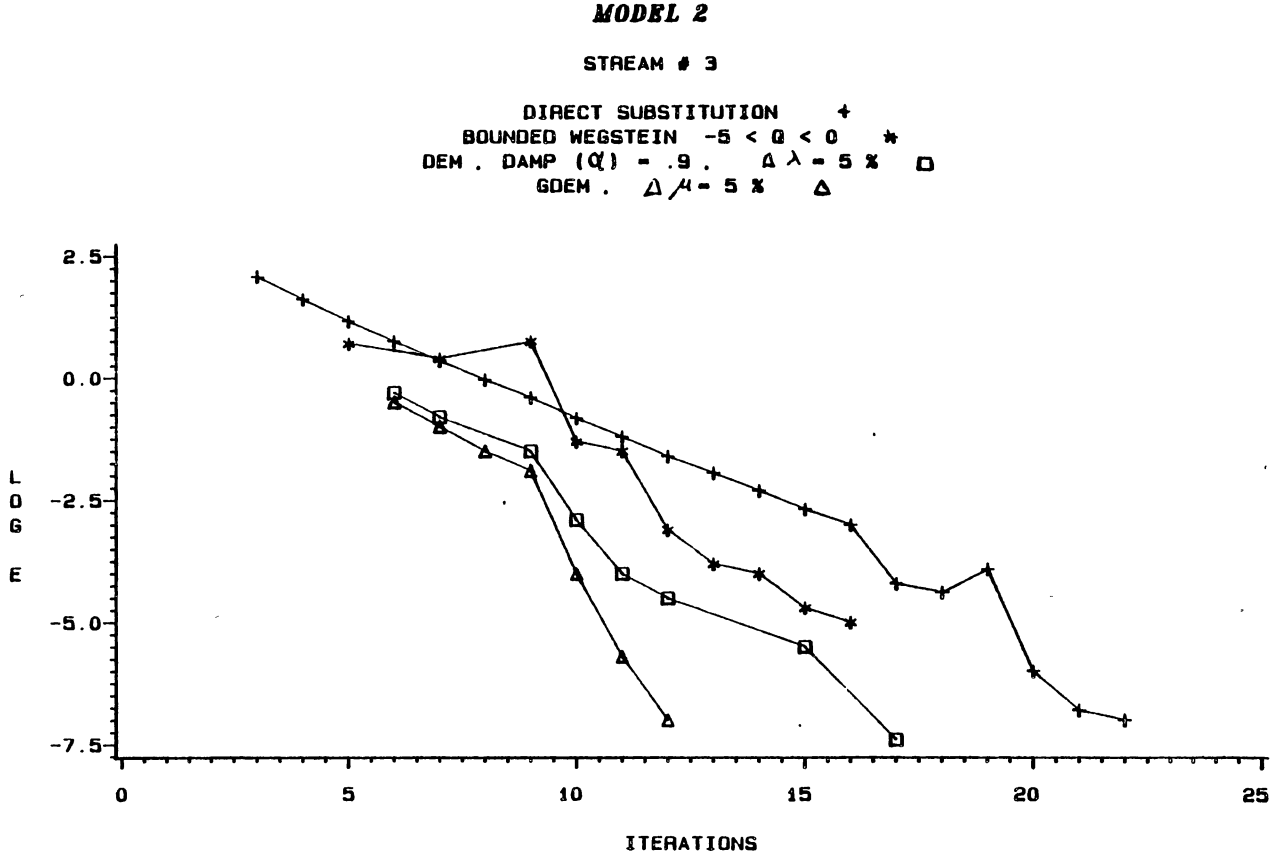


Figure 6. Comparison of the Three Methods in Model 2

attempted before the eigenvalue has stabilized. Consequently the estimate will be less accurate as opposed to that from a smaller $\Delta\tilde{\mu}$ which means the acceleration will be delayed several iterations but the estimate will be more accurate. This is a question of trade off which will be discussed later in more detail.

In Chapter I some of the individual variables mentioned which could be used in an acceleration algorithm were the individual mass flow rates, the system temperature, pressure, quality, enthalpy, and entropy , etc.. Gibbs theorem (12) states that all the properties of a system are completely determined given the composition and two independent variables in the system. The simulation package, MAXISIM has a built in flash operation that can determine the system's condition completely given the composition, temperature, and pressure. Since all process unit operations on MAXISIM are performed isobarically (i.e. at constant pressure) this leaves only one independent variable that could be used in the acceleration algorithm. Therefore, the logical choice for the individual variables in the acceleration algorithm were the individual flow rates and the stream temperature. Note that to have increased the variables in the acceleration by another independent variable would have over defined the system causing thermodynamic inconsistencies.

The logical steps of how a typical acceleration algorithm like GDEM would interact with the main simulation software

is shown in the form of a flowchart in APPENDIX D. The composition and the conditions of the feed streams to the chemical simulations are listed in APPENDIX E.

At this point several questions needed to be answered. 1) How to choose the stream to be accelerated ? 2) Is there an optimum value for $\Delta\hat{\mu}$? , and 3) How GDEM would perform against a more oscillatory system ?

Table I shows the results of Model 2 computations. The best result was a convergence in 11 iterations with $\Delta\hat{\mu}$ = 1%. Almost a 50% reduction in number of iterations and 45 % reduction in computer time over direct substitution. Six more models each with some specific characteristics were chosen to further test GDEM .

Model 3 is basically a heat exchanger dominated system where only heat is transfered to the feed stream (Figure 7). Although Model 3 shows no oscillation, very little mass is recycled to the heat exchanger reducing the effect of the acceleration resulting in only a 15% reduction in number of iterations (Table II).

Model 4 is a combination of heat and mass transfer dominated system where only one stream is recycled through the heat exchanger and another recycled through a flash operation (Figure 8). This model shows a surprising degree of oscillation such that acceleration could not be attempted resulting in no improvement over D.S. (Table III).

TABLE I

RESULTS OF THE GDEM ALGORITHM IN MODEL 2

STREAM #	$\Delta \hat{\mu}$ (%)	ITERATIONS
2	5	13
2	1	11
3	5	12
3	1	11
4	5	13
6	5	21
7	5	14
10	5	13
10	1	11

Tolerance = 1E-4

Direct Substitution = 21 Iterations

Method = GDEM

MODEL 3

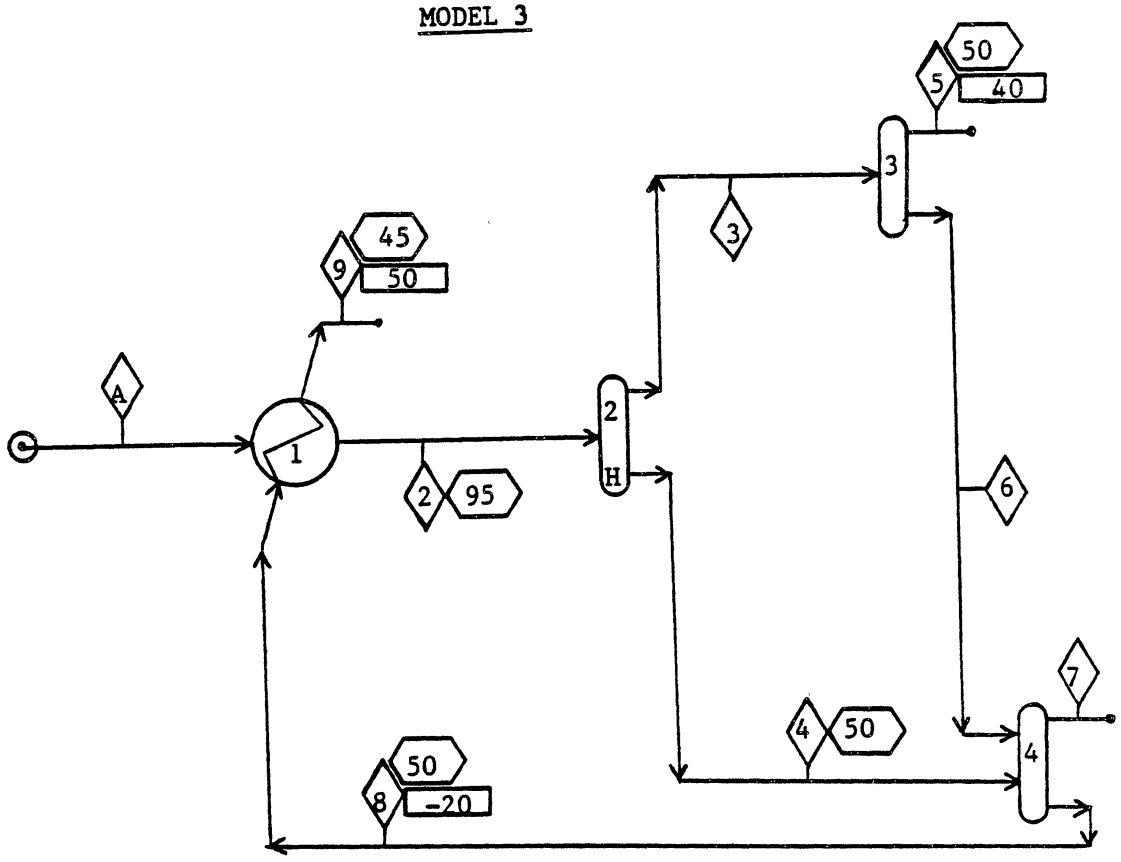


Figure 7. A Heat Exchange Dominated System

TABLE II

RESULTS OF THE GDEM ALGORITHM IN MODEL 3

STREAM #	$\Delta\hat{\mu}$ (%)	ITERATIONS
2	5	11
8	5	11

Tolerance = 1E-5

Direct Substitution = 13 Iterations

Method = GDEM

MODEL 4

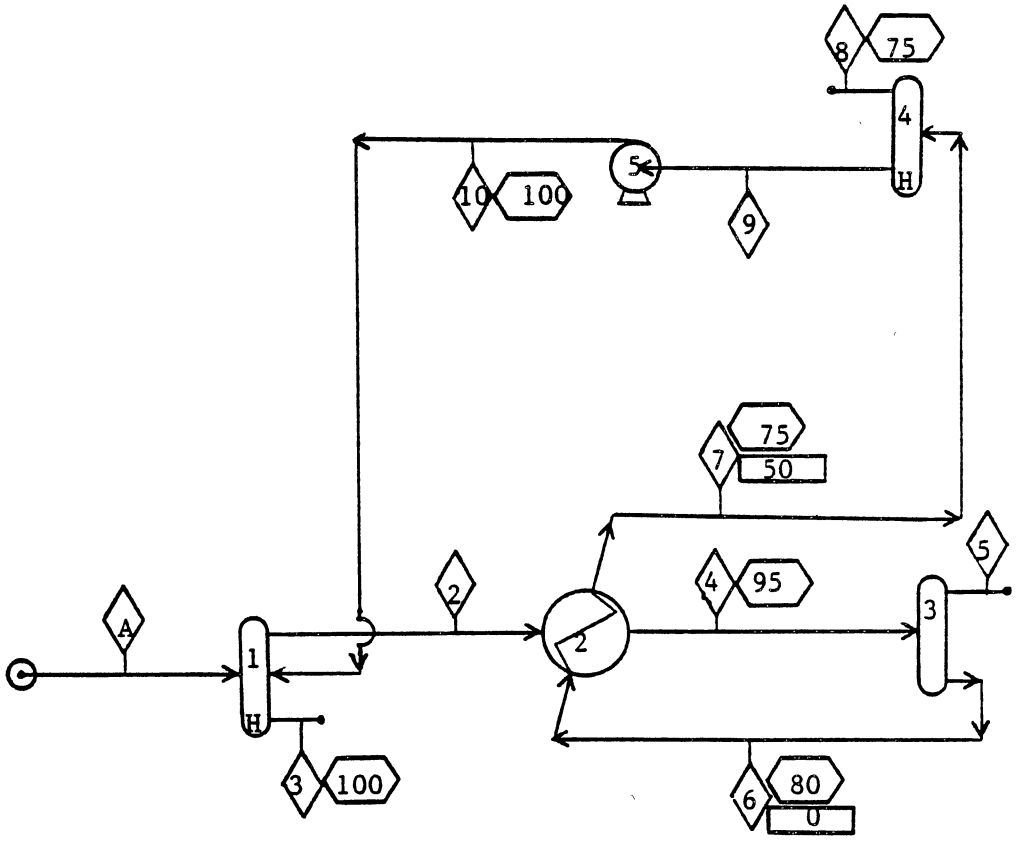


Figure 8. A Heat and Mass Dominated System

TABLE III

RESULTS OF THE GDEM ALGORITHM IN MODEL 4

STREAM #	$\Delta\hat{\mu}$ (%)	ITERATIONS
2	5	14
4	5	14
6	5	14
10	5	14

Tolerance = 1E-5

Direct Substitution = 14 Iterations

Method = GDEM

Model 5 is a stream divider dominated system with only one recycle stream through a flash operation (Figure 9). A stream divider is the simplest form of an unit operation where all the characteristics of the stream remain intact while the mass flow rate is divided. In Model 5, the stream has been divided into a 10% to 90% ratio in mass flow rate. The non-oscillatory behavior of this model lends itself very nicely to acceleration algorithms. The results of Model 5 calculations are tabulated in Table IV with almost a 90% reduction in the number of iterations compared to direct substitution.

Model 6 is a purely flash dominated system with only one recycle stream (Figure 10). Also a very typical chemical process model, the best results for Model 6 were obtained accelerating the recycle stream and $\Delta\hat{\mu}$ = 10% with a convergence in 11 iterations with a 50% saving in the number of iterations (Table V).

Model 7 introduces a distillation column connected to three flashes with one recycle stream (Figure 11). This model like Model 4 showed a surprising degree of oscillation which translates into zero improvement in convergence over D.S. (Table VI). Note that Process Model 7 is not a realistic representation of a process system and is introduced here purely for academic reasons to test GDEM against oscillatory systems.

Model 8(A) is a more complicated system with 6 recycle

MODEL 5

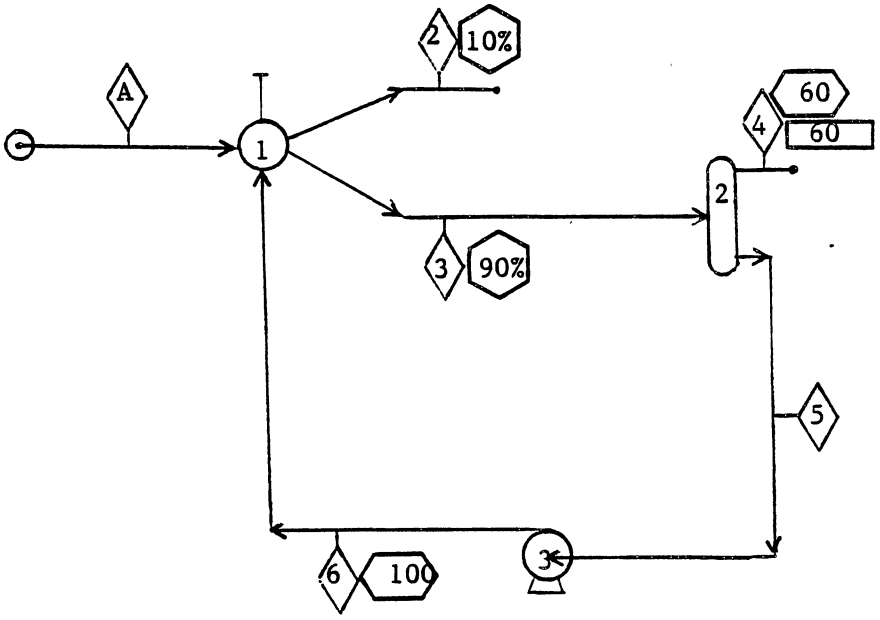


Figure 9. A Stream Divider Dominated System

TABLE IV

RESULTS OF THE GDEM ALGORITHM IN MODEL 5

STREAM #	$\Delta\hat{\mu}$ (%)	ITERATIONS
3	5	10
3	1	10
6	5	12
6	1	12

Tolerance = 1E-5

Direct Substitution = 90 Iterations

Method = GDEM

MODEL 6

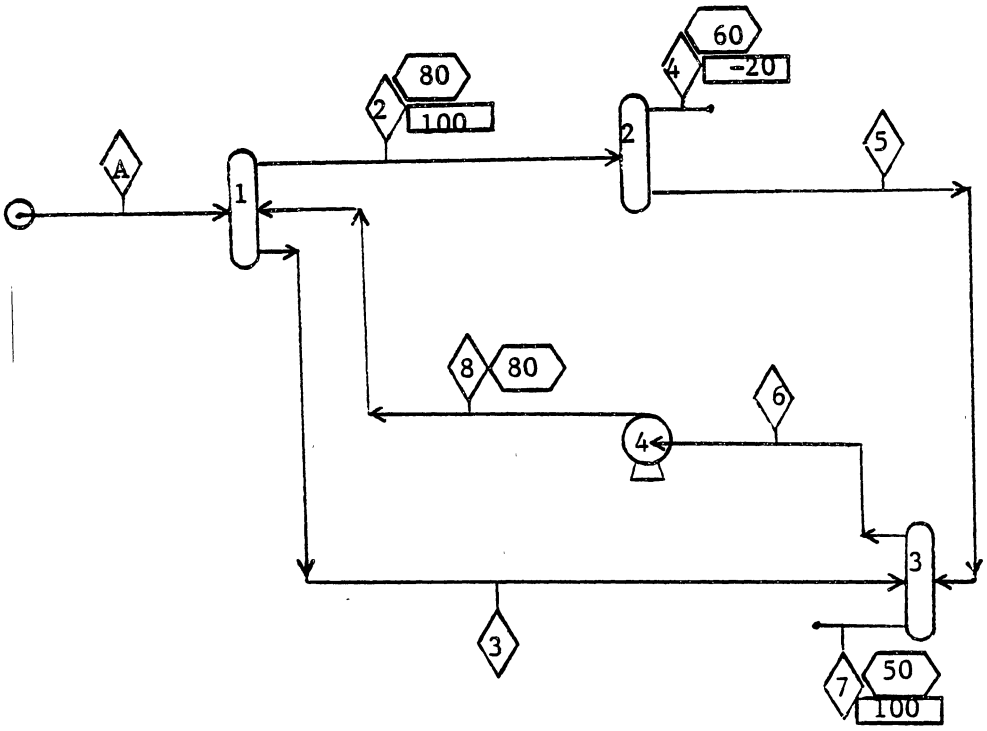


Figure 10. A Flash Dominated System

TABLE V

RESULTS OF THE GDEM ALGORITHM IN MODEL 6

STREAM #	$\Delta\hat{\mu}$(%)	ITERATIONS
2	5	18
3	5	21
8	1	14
8	5	13
8	10	11

Tolerance = 1E-5

Direct Substitution = 22 Iterations

Method = GDEM

MODEL 7

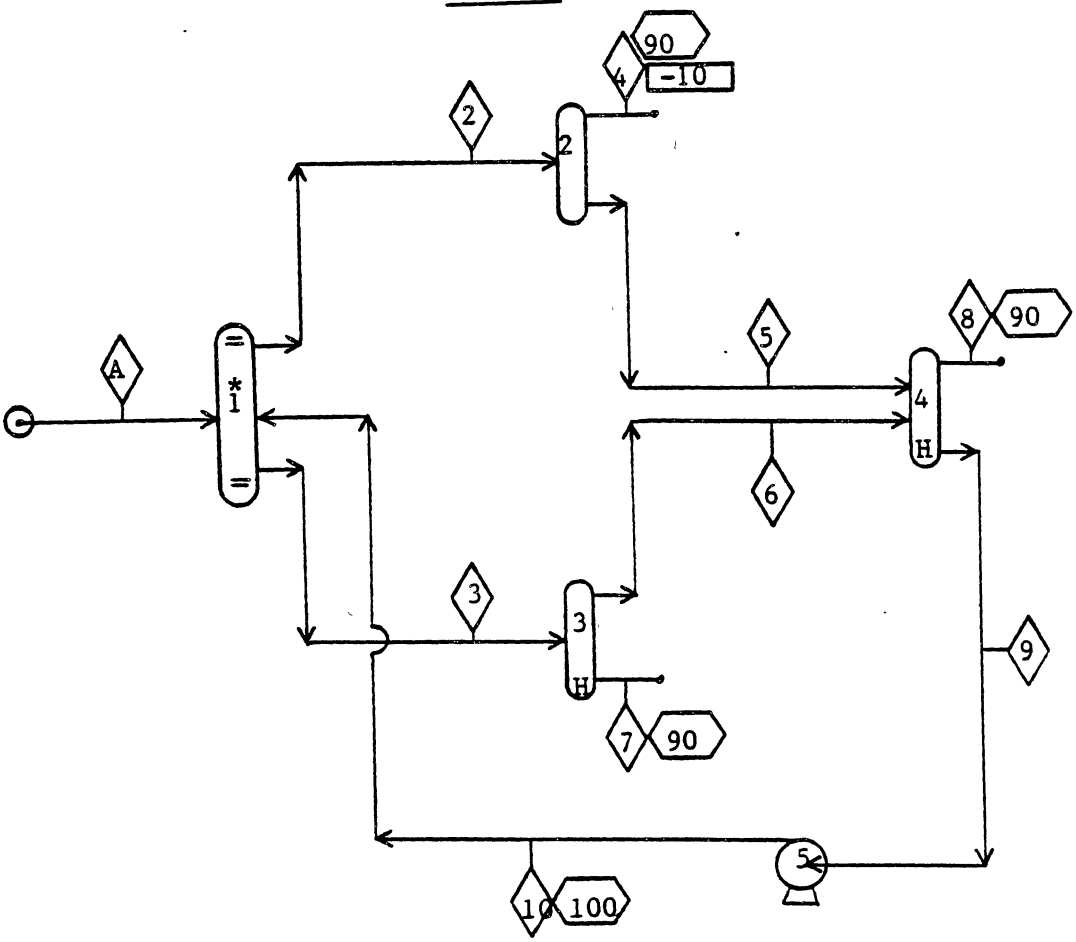


Figure 11. System Containing a Distillation Column

* APPENDIX F

TABLE VI

RESULTS OF THE GDEM ALGORITHM IN MODEL 7

STREAM #	$\Delta\hat{\mu}$ (%)	ITERATIONS
10	5	15
10	1	15

Tolerance = 1E-5
Direct Substitution = 15 Iterations
Method = GDEM

MODEL 8(A)

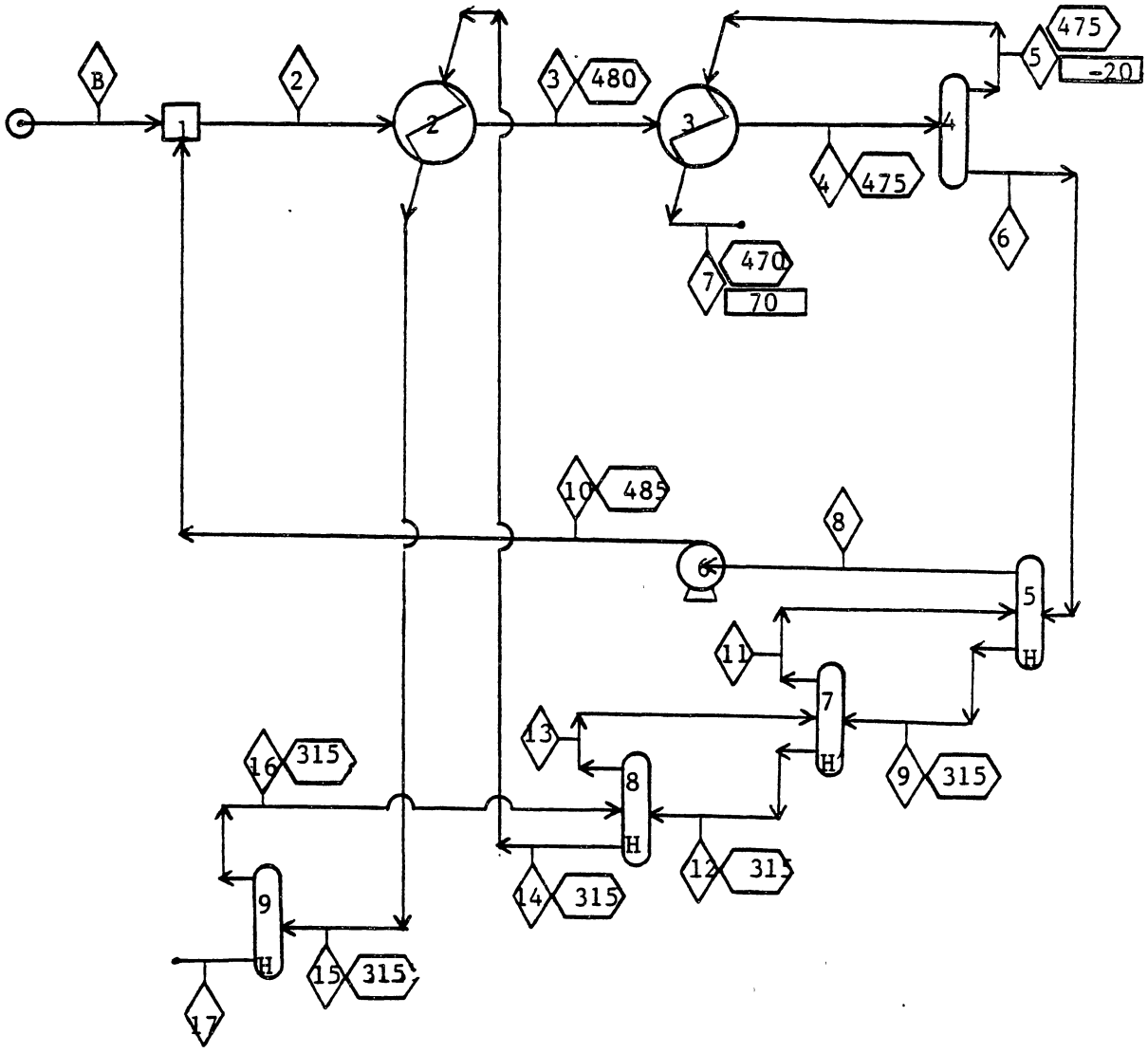


Figure 12. System With Four Flash Drums in Series

streams (Figure 12). This model would normally be used in the latter stages of a typical design process. A closer look at Model 8(A) will reveal that units 5, 7, 8, and 9 are connected in series simulating a distillation column where the heat source for the reboiler duty is supplied by feed stream #2 at heat exchanger unit 2. It was not very surprising to find that this model also was very oscillatory because of the high degree of complexity of the model (Table VII). Figure 13 shows the oscillatory behavior of stream #4 in Model 8(A).

The assumption of linearity near the solution is an important condition for acceleration algorithms. In the case of Model 8(A) this condition was not met therefore it is no surprise that the % change in eigencoefficient, $\Delta\tilde{\mu}$, was very unstable which means that no acceleration could have been attempted.

Flash units 5, 7, 8, and 9 can be replaced with a distillation column of equal characteristics (i.e. the same separation of the light and heavy key components) and heat exchanger unit 2 can be replaced with a heater/cooler unit removing heat of equal duty as the reboiler in the distillation column. Thus Model 8(B) is created from Model 8(A) where the distillation column has replaced the flash operations 5, 7, 8, and 9 (Figure 14). With D.S. Model 8(B) converges in only 8 iterations as opposed to 33 for Model 8(A). The quick convergence of the system renders the acceleration algorithm useless (Figure 15). However, note that the

TABLE VII

RESULTS OF THE GDEM ALGORITHM IN MODEL 8(A)

STREAM #	$\Delta\hat{\mu}$ (%)	ITERATIONS
10	1	33
16	1	33

Tolerance = 1E-4

Direct Substitution = 33 Iterations

Method = GDEM

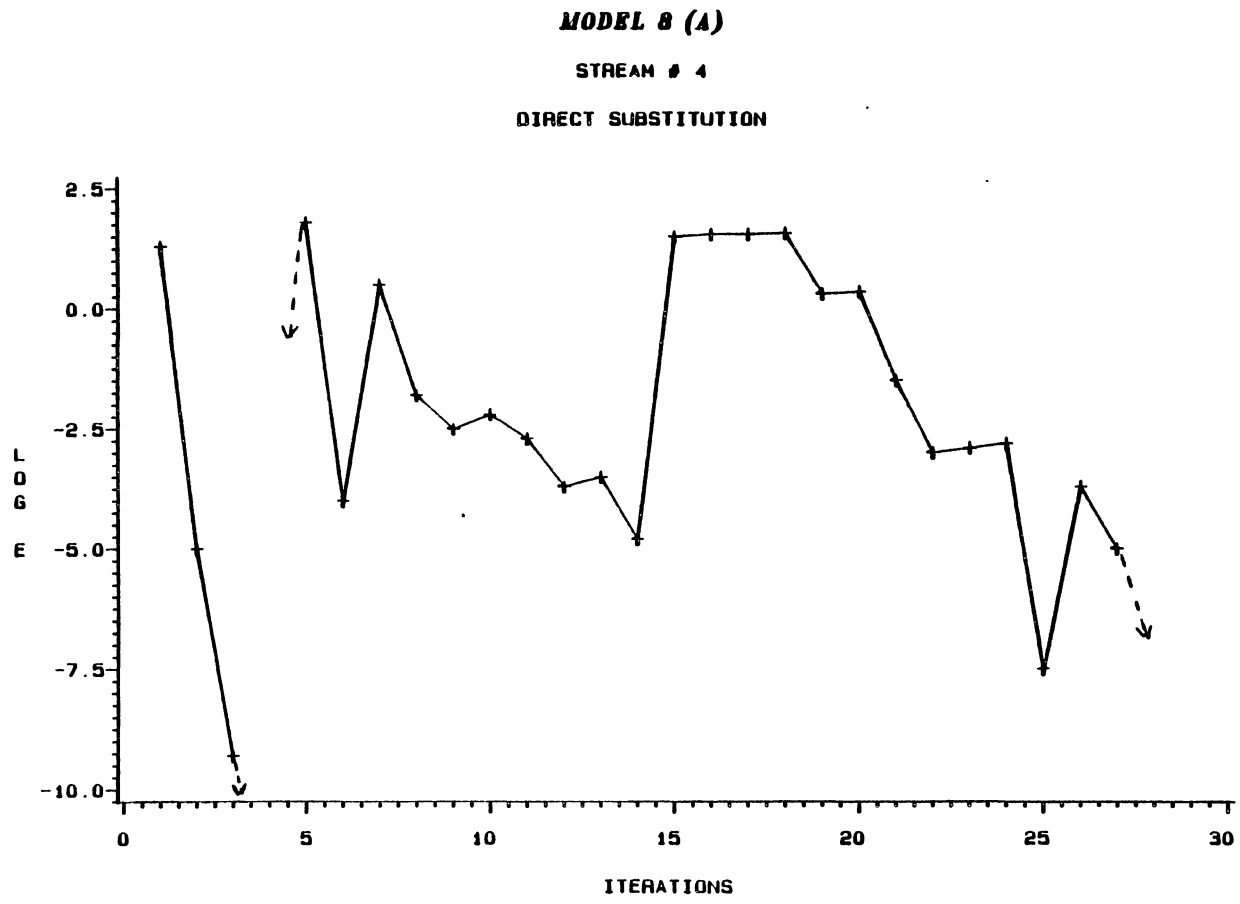


Figure 13. Oscillatory Behavior of Flash Drums in Series

MODEL 8(B)

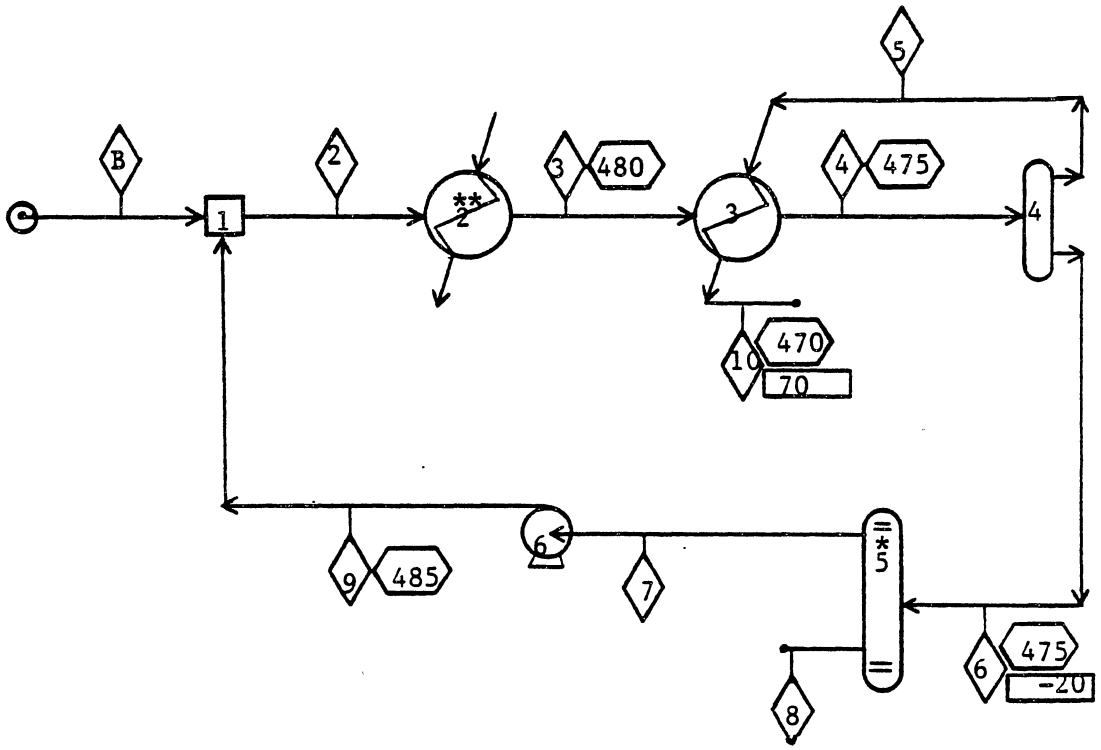


Figure 14. Distillation Column Replacing Flashes in Series

* APPENDIX G

** -112.43 KBTU/Hr

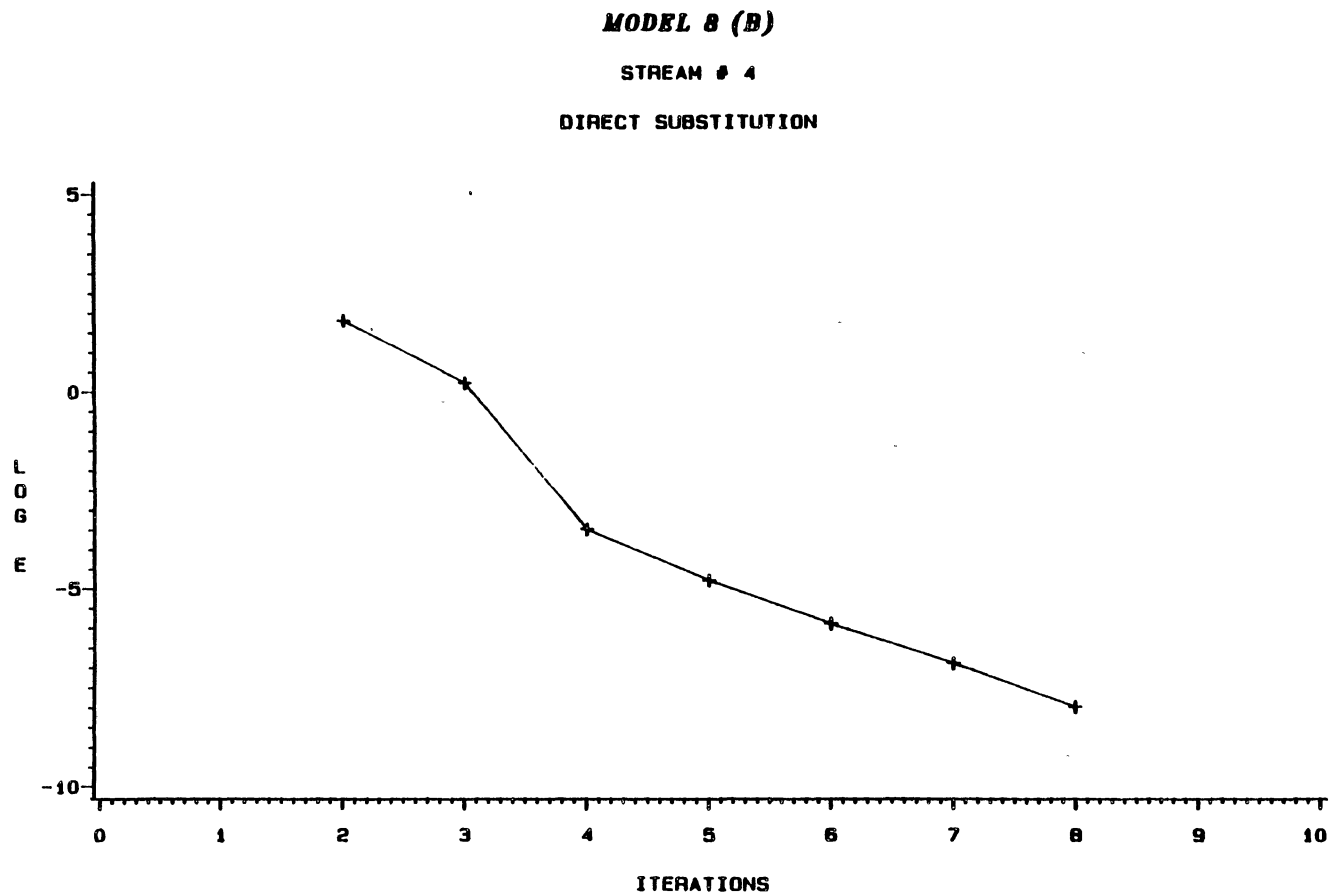


Figure 15. Stability of Stream # 4 in Model 8 (B)

information about the reboiler duty and the column characteristics about Model 8(B) were known only after Model 8(A) had converged to a solution. although Model 8(A) and 8(B) have similar separation characteristics, Model 8(B) shows much less oscillation because the flash operations imbeded in the distillation column converge iteratively before the next operation can be performed. Secondly heater/coolers have a fixed heat load unlike heat exchangers that can have variable heat loads that change with changing flow rates.

It was found from the example process models that there are no clear cut criteria to determine the best stream for acceleration. In most cases the outermost recycle stream worked best. One exception to this rule was found in process model 5 where a branched stream had a slight advantage over the recycle stream. Note that the roles could easily have been reversed in favor of the recycle stream if some of the preset conditions in the system were changed. Also in most cases a value of $\Delta\hat{\mu}$ = 1 % seemed to work best although again an exception was found in process model 6 where $\Delta\hat{\mu}$ = 10 % gave the best results.

The reduction in cpu time (computer time) is not always proportional to the reduction in the number of iterations. As the system approaches convergence the cpu time for each iteration usually decreases. Therefore, the actual saving in computer time will always be a little less than the proportional reduction in the number of iterations. For example, as was said earlier in Model 2 for a 50 % reduction in

the number of iterations, the actual reduction in cpu time was only 45 %. However, the reduction in iterations should serve as a good indicator for the actual saving in cpu time. It should also be said that including the temperature as an independent variable in the acceleration does not constitute a significant saving in the number of iterations. However, it should not slow convergence either because in all the systems studied the stream temperature has converged rapidly and did not dominate the calculations of the eigenvalue. Note that if for some reason the temperature in an hypothetical process did not stabilize quickly in a stream with a small flow rate (ie. 10 total moles/hr) the shear size of the temperature (ie. 200 deg.F) could actually dominate the eigenvalue and maybe even slow convergence.

CHAPTER IV

SUMMARY, CONCLUSIONS, AND RECOMMENDATIONS

Summary and Conclusions

In a mixture of chemical compounds the interaction between the different components is very complex and nonlinear. Wegstein's method becomes ineffective as a tool for acceleration because each individual component is treated independently of others. Furthermore, there are no logical criteria for when the acceleration should be applied. It has been shown that much better results can be obtained if the stream to be accelerated is treated as a matrix rather than a series of unknown equations. DEM and GDEM are methods of acceleration based on the eigenvalue of this matrix. The improvement over Wegstein is two fold. 1) The interaction among the components is considered. 2) The criterion for acceleration is based solely on the stability of the eigenvalue.

The best results were for GDEM ranging from zero improvement for a very oscillatory system to over 90 % reduction in the number of iterations in the case of an non-oscillatory system. Model 2 which represented a typical chemical process had a reduction of almost 50 % in the number of iterations.

To understand why there is such a large difference in improvement from one system to another, it must be first understood what makes one system more oscillatory than another. Of course if the system were understood completely there would be no need for an acceleration algorithm. However, there are several guidelines that can help in understanding this oscillatory behavior. For example, oscillation usually increases with increasing number of recycle streams in the system and/or if the system contains complex operations like distillation towers as opposed to simple operations like stream dividers. Other guidelines are more subtle, like how the unit operations are arranged and the system preset conditions.

Automation in the acceleration program can be achieved if a suitable stream can be chosen for acceleration with an appropriate value for $\Delta\hat{\mu}$. The distinction between the advantages in acceleration of one stream over another is usually based on a prior knowledge or experience with process systems. Without such prior knowledge the outermost recycle stream can be chosen as a suitable candidate for acceleration. It was also found that a value of 1 % for $\Delta\hat{\mu}$ worked best for most systems. Higher values for $\Delta\hat{\mu}$ can be chosen only at the risk of oscillating the system at each acceleration.

In conclusion the best results were based on the GDEM using a recycle stream for acceleration with an acceleration criterion of $\Delta\hat{\mu}$ = 1 % achieving a reduction of 50 % in the

number of iterations which approximately corresponds to a 45 % reduction in cpu time. Therefore, we highly recommend the use of the GDEM algorithm to significantly reduce the computer usage and cost. We also found GDEM to be highly suitable and effective as an acceleration algorithm for process design simulations. It's use is also not confined to the convergence of process systems but can also be used anywhere a convergence parameter is needed to be determined iteratively requiring ten or more iterations, a common characteristics for many chemical equilibria calculations.

Recommendations for Further Study

1. Quasi-Newton Methods can be as an alternative to the GDEM if methods for updating the Jacobian, starting with the identity matrix can be improved. Soliman (13) recommends that the convergence can be improved if Pn = -F(Xn) and Ho = I.

2. Convergence may be improved if instead of starting the Jacobian as Ho = ∂ F(Xn) / ∂ (Xn) or Ho = I, a partially determined Ho is used based on a certain criterion. This criterion might be the highest mass percent of the components in the stream or the components in the mid-range between the lightest and the heaviest components.

LITERATURE CITED

1. Aitken, A.C., "On Bernouli's Numerical Solution of Algebraic Equations," Proc. Royal Soc. Edinburgh, 46, 289-305 (1925).

2. Broyden, C.J., "A Class of Methods for Solving Nonlinear Simultaneous Equations," Math. Comp., 19, 577-593 (1965).

3. Carnahan, B., Luther, H.A., and Wilkes, J.O., "Applied Numerical Methods," John Wiley, 5, 310-320 (1969).

4. Conte, S.D., and Boor, C.D., "Elementary Numerical Analysis: An Algorithmic Approach," MacGraw - Hill Book Company, 79, 223-226 (1980).

5. Crowe, C.M., and Nishio, M., "Convergence promotion in the Simulation of Chemical Processes - the General Dominant Eigenvalue Method," AICHE J., 21, 528-533 (1975).

6. Erbar, J.H., Revised by Erbar, R.C., "MAXISIM," SCI 2701 Fox Ledge Lane, RR5, Stillwater, OK. 74074, Tel.(405) 377-4279 (1984).

7. Faddeeva, V.N., "Computational Methods of Linear Algebra," Dover Publications, Inc., New York, 60-62 (1959).

8. Graves, T.R., "An Evaluation of Convergence Acceleration Methods for Chemical Process Recycle Calculations," Ph.D. Thesis, Oklahoma State University, Stillwater (1972).

9. Kliesch, H.C., "An Analysis of Steady State Process Simulations : Formulation and Convergence," Ph.D. Thesis, Tulane University, New Orleans, La. (1967).

10. Noble, B., "Applied Linear Algebra," Prentice - Hall, Englewood Cliffs, N.J., 372-374 (1969).

11. Orbach, O., and Crowe, C.M., "Convergence Promotion in the Simulation of Chemical Processes With Recycle - the Dominant Eigenvalue Method," Can. J. Chem. Eng., 49, 509-513 (1971).

12. Smith, J.M., and Van Ness, H.C., "Introduction to Chemical Engineering Thermodynamics," Third Edition, MacGraw - Hill Book Company, 39-40 (1975).

13. Soliman, M.A., "Quasi-Newton Methods for Convergence Acceleration of Cyclic Systems," Can. J. Chem. Eng., 57, 643-647 (1979).

14. Wegstein, J.H., "Accelerating Convergence of Iterative Processes," Comm. Assoc. Comput. Mach., 1 (6), 9-13 (1958).

APPENDIXES

APPENDIX A

BEHAVIOR OF THE MATRIX NEAR THE SOLUTION

The assumption of linearity near the solution is a critical assumption for convergence of a system. As an example, Figure 16 shows the total mass of stream # 3 in model 2 (See model 2 discussion on page 17). The total mass at a given iteration is divided by the final mass for easy comparison.

As can be seen the total mass follows a predictable curve and is non-oscillatory while approaching linearity near the solution. However, not all process simulations are non-oscillatory. This is specially true in a series of flashes with connecting recycle streams simulating a distillation tower. Figure 17 showes such an oscillatory behavior in stream # 4, model 8. Note that although it may appear that the stream is approaching linearity near the solution ,in actuality the oscillation still exists in a smaller scale until the whole system converges at iteration # 30.

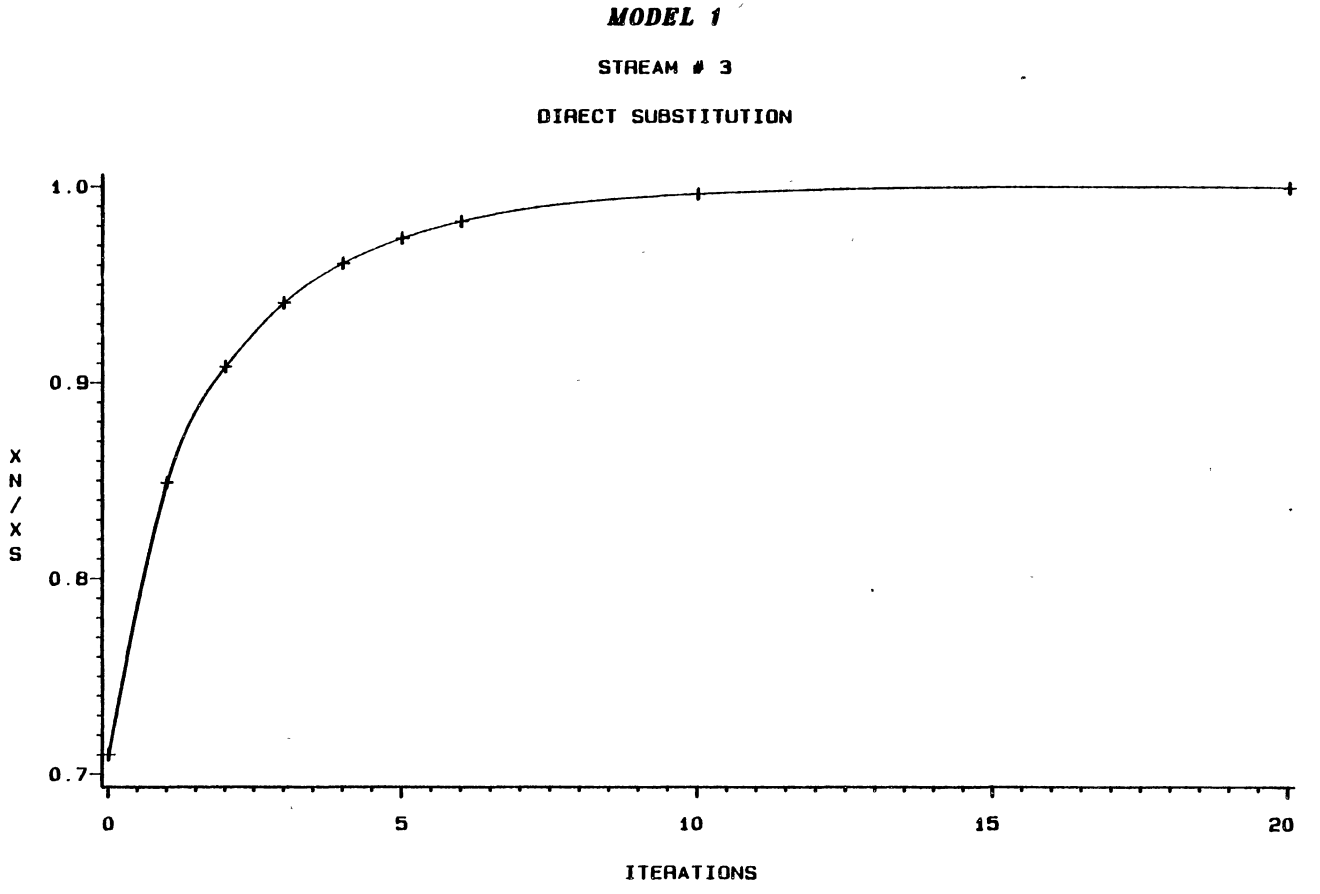


Figure 16. History of the Sum of the Mass of Stream # 3

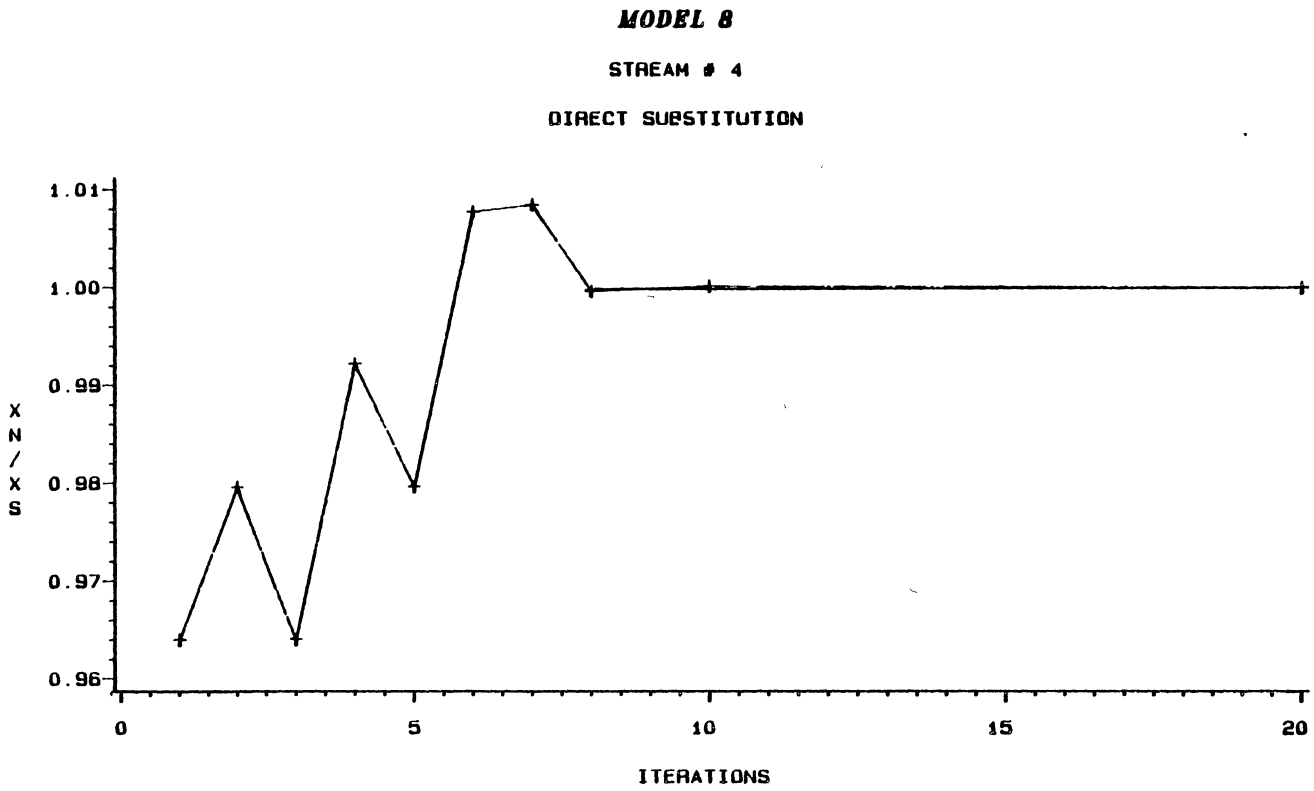


Figure 17. Oscillatory History of the Sum of the Mass of Stream # 4

APPENDIX B

OPTIMIZING THE PERFORMANCE OF THE WEGSTEIN'S METHOD

In the optimization of the Wegstein's method model 2 was used throughout the test. The optimization of the bounded Wegstein was made by trying to answer the following three questions. 1)What is the best range for q ? 2)How does including the temperature of the stream affect the convergence of the matrix ? And finally 3)Will applying the Wegstein at every iteration help improve convergence ?

Figures 18, 19, and 20 show that for model 2 the best range for q is between -5 and 0 . They also show that including the temperature as a variable in the matrix and applying Wegstein every other iteration will help improve the convergence of the method.

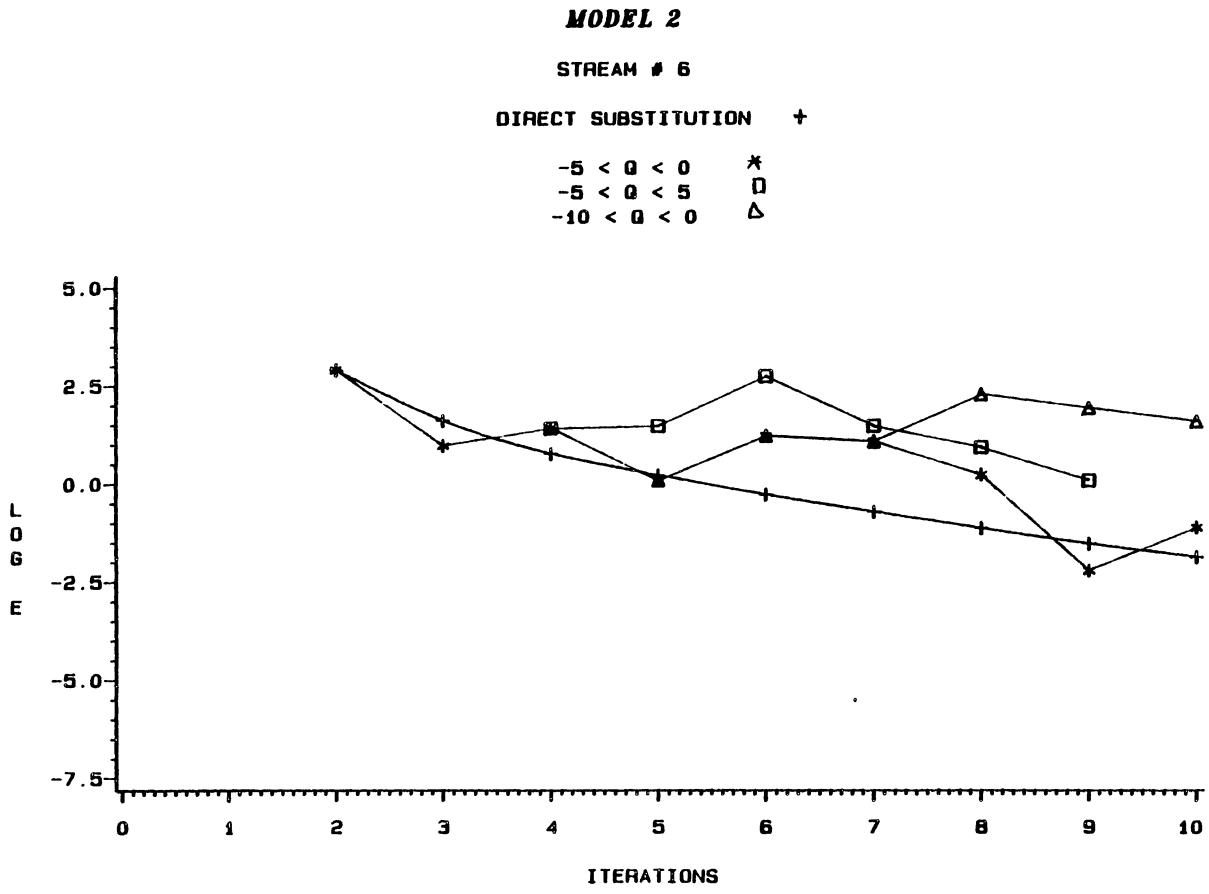


Figure 18. Optimizing q in the Bounded Wegstein

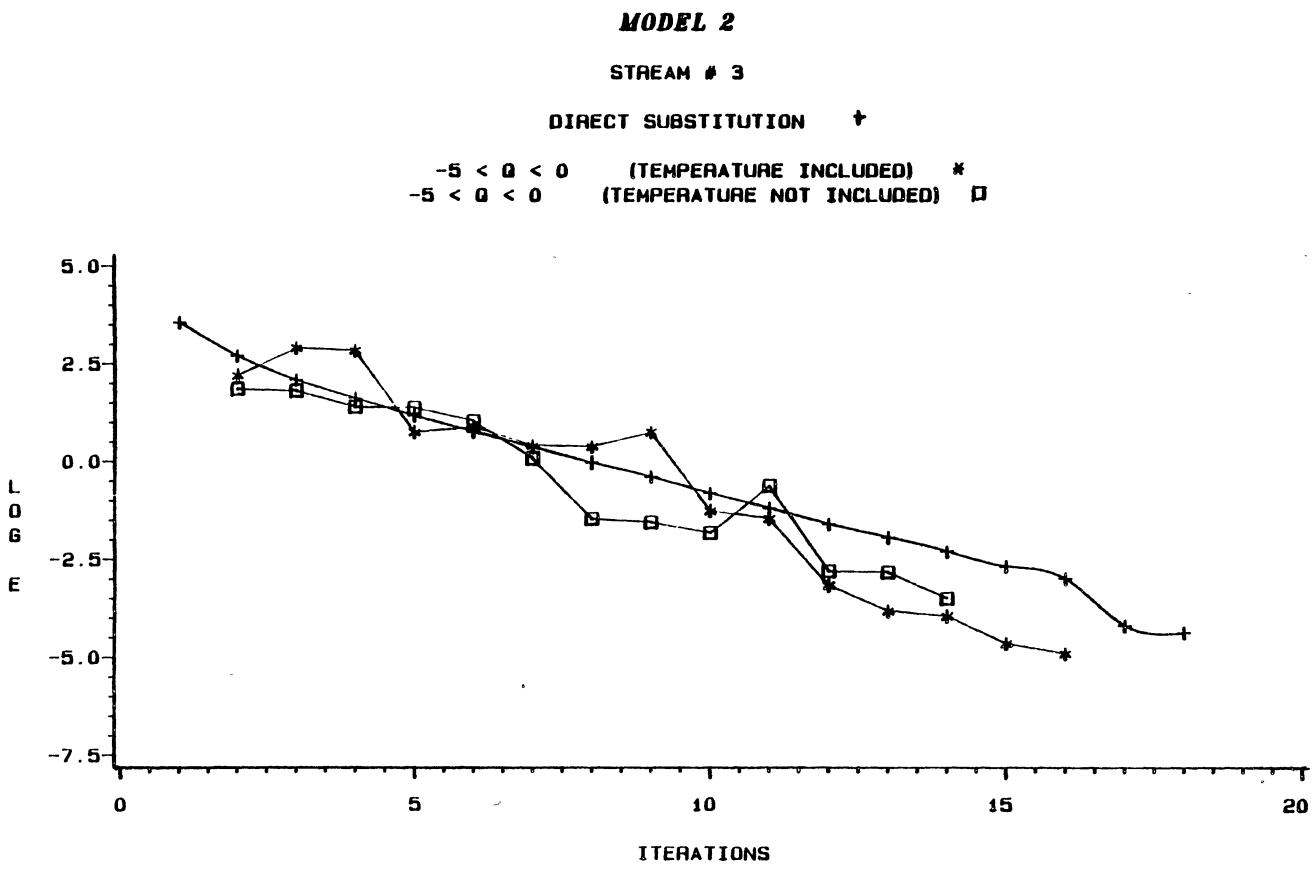


Figure 19. The Effect of Temperature on the Convergence of Wegstein

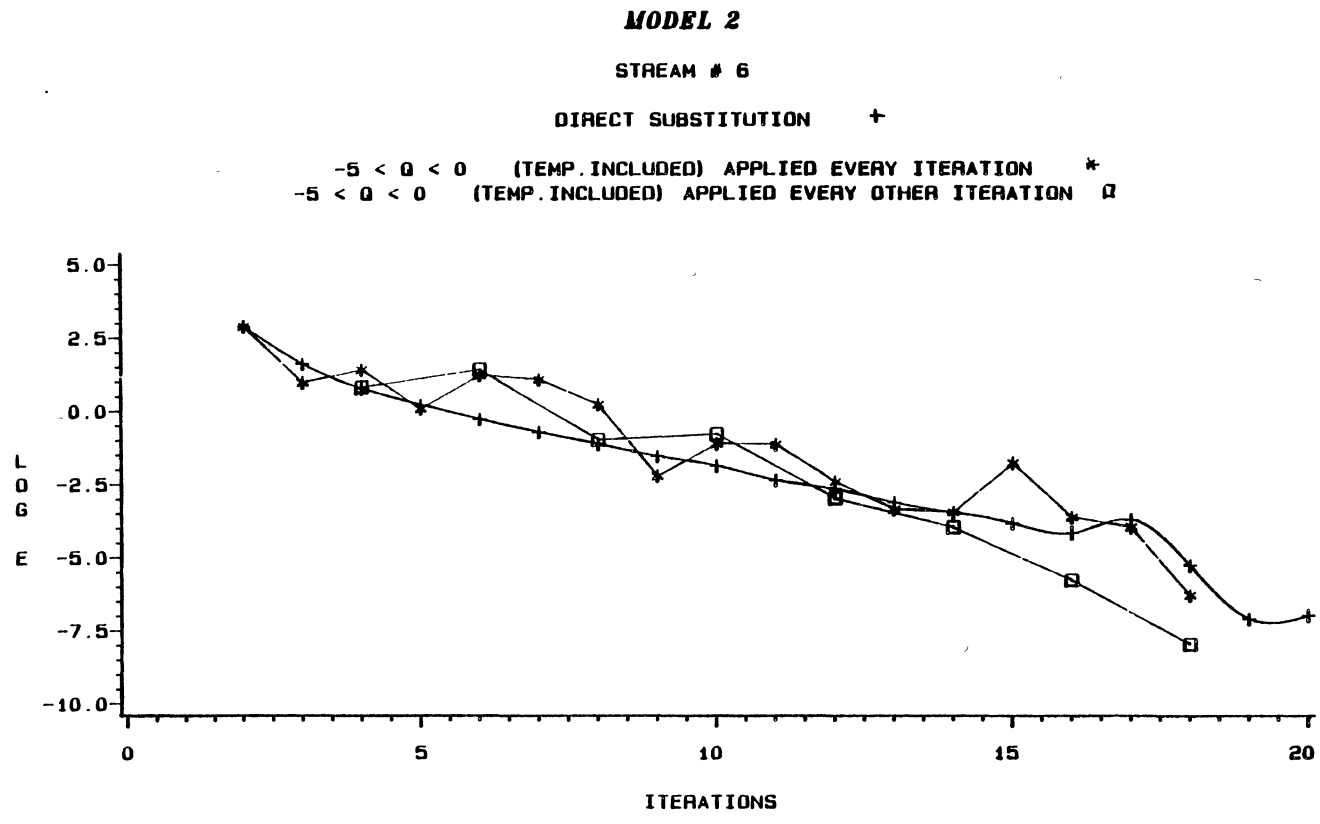


Figure 20. Wegstein Applied Every Other Iteration

APPENDIX C

OPTIMIZING THE DEM

The Orbach and Crowe's DEM was optimized based on stream # 3 in model 2. Figure 21 showes no significant difference for $\Delta\lambda$ in the range of 2 to 20 %. In Figure 22 a damping factor of 0.9 shows a slight improvement in convergence. These findings are in agreement with the findings of Orbach and Crowe (11) .

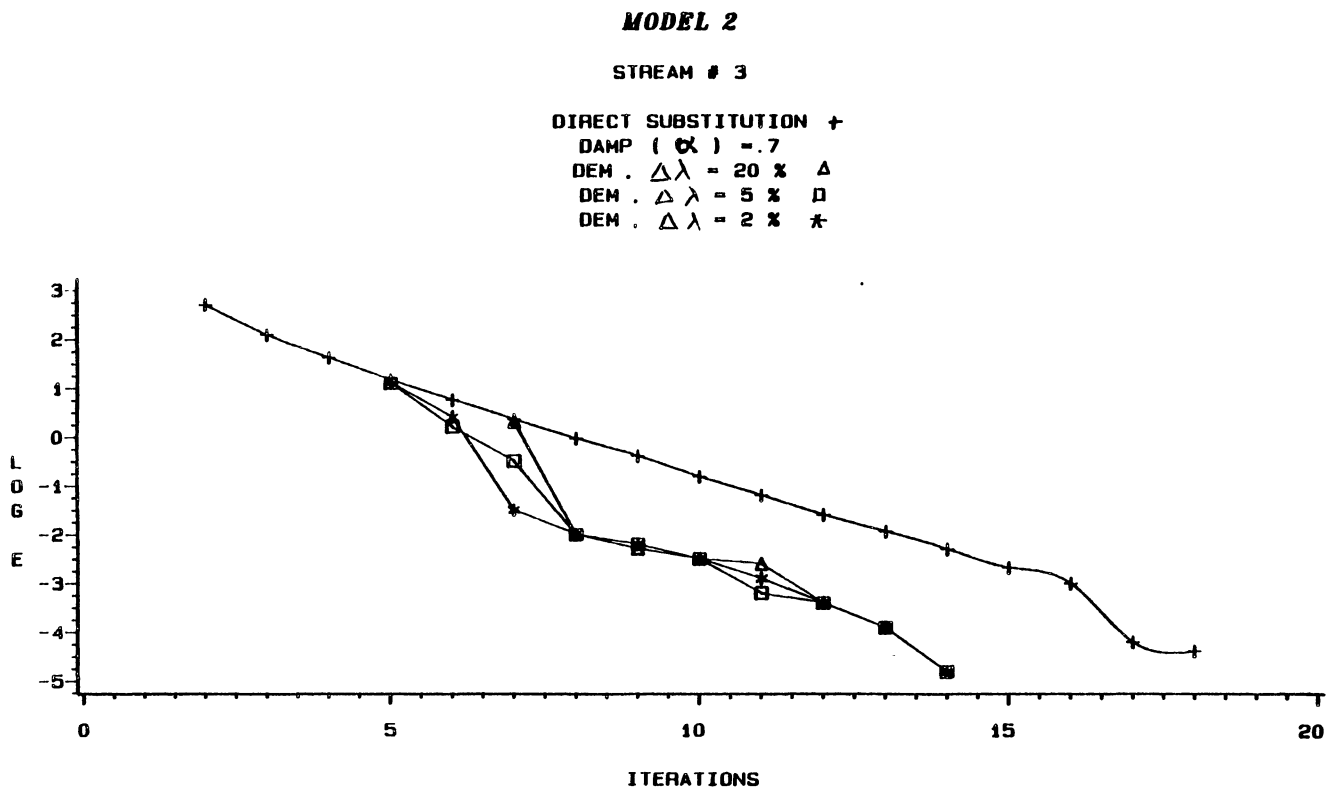


Figure 21. Orbach and Crowe's DEM Applied at Different $\Delta\hat{\lambda}$

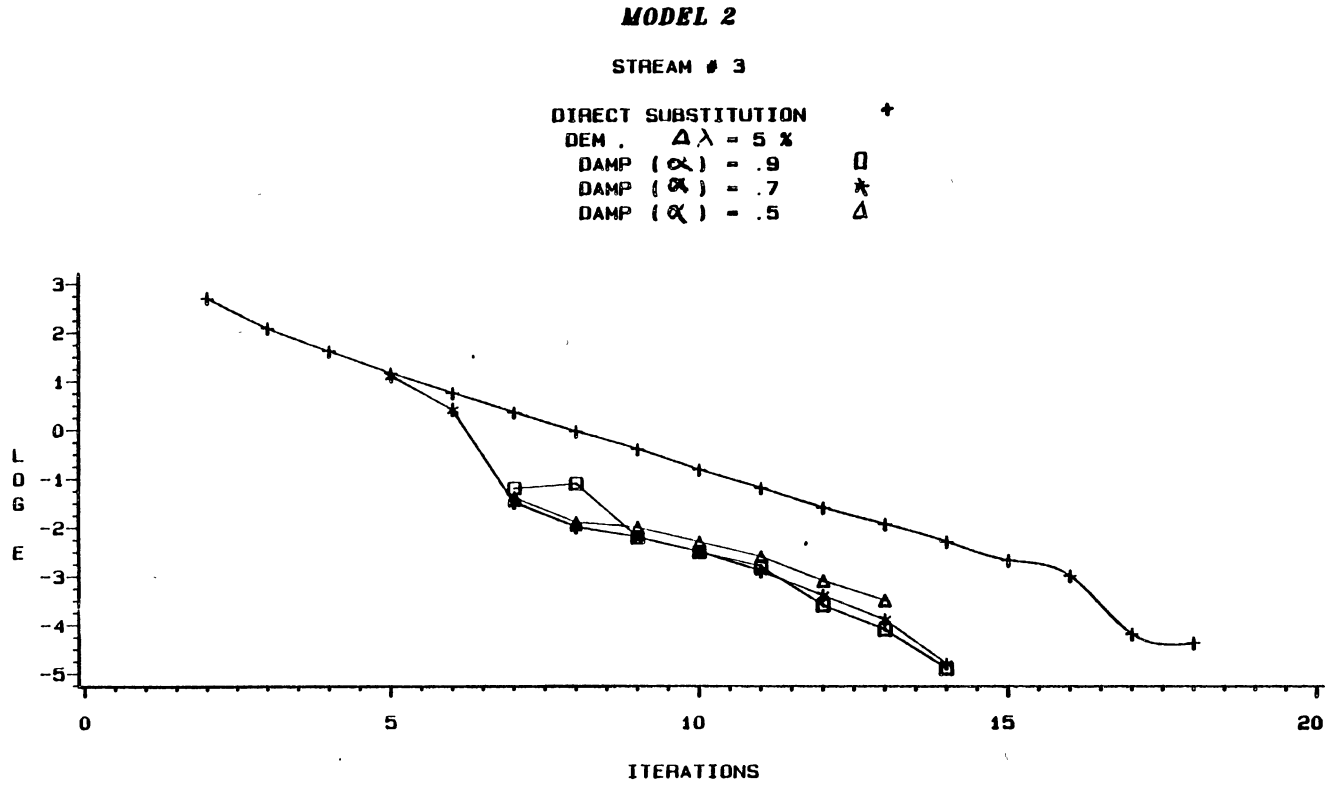


Figure 22. Orbach and Crowe's DEM Applied at Different Damping Factors

APPENDIX D

FLOWCHART OF THE GDEM ACCELERATION ALGORITHM

The following flowchart shows how the series of logical steps are taken that determines if, how, and when the GDEM acceleration should be applied (Figure 23) . The first step is the execution of all unit operations in series. Next if a recycle stream exists then an iterative procedure would be followed, otherwise the results are printed and the program stops. If the pressure is zero very likely the stream is empty and a warning statement is printed. The contents of the stream to be accelerated are stored in an array. Three sets of arraies will be required to store the pressure, temperature, and the stream composition. In the third iteration the criteria for the acceleration is checked. If $\Delta\hat{\mu}$ < 1 % and 0 < $\hat{\mu}$ 2 < 1 are true then the acceleration is attempted. Stepping back the array is needed to discard the old stream and enter the new stream values. The flash operation is necessary after each acceleration to correct the quality and other thermodynamic properties of the stream. The simulation is again checked for convergence. If it has not converged the cycle is repeated until convergence is achieved.

Acceleration Flowchart

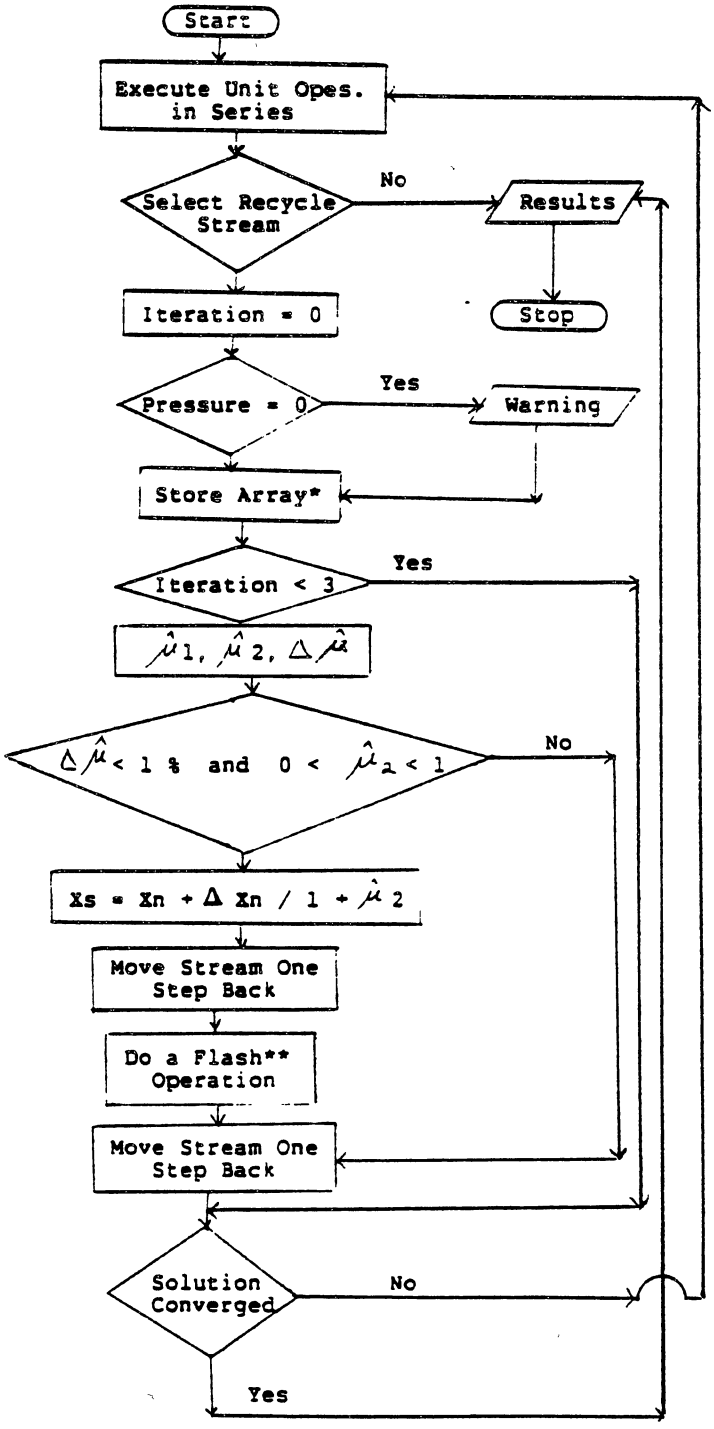


Figure 23. Flowchart for the GDEM Algorithm

APPENDIX E

COMPOSITION AND CONDITIONS OF THE INPUT STREAMS IN THE PROCESS MODELS

TABLE VIII

FEED STREAMS TO THE EXAMPLE MODELS

FEED A (lbmole/hr)	FEED B (lbmole/hr)
C2H6 = 100	N2 = 3.47
N-C4H10 = 80	CH4 = 204.75
N-C5H12 = 60	CO2 = .554
N-C6H14 = 40	C2H6 = 24.59
temperature = 100 (F)	C3H8 = 17.47
pressure = 100 (psia)	IC4H10 = 3.457
	NC4H10 = 5.224
	IC5H12 = 1.689
	NC5H12 = 1.214
	NC8H18 = 1.4776
	temperature = 100 (F)
	pressure = 485 (psia)

APPENDIX F

CHARACTERISTICS OF THE DISTILLATION UNIT IN MODEL 7

```
DTXT #  2
   COUNT PLATES FROM BOTTOM UP
   NUMBER OF PLATES IN COLUMN     13
   NUMBER OF FEED PLATES           2
   NUMBER OF PRODUCTS              2
   NUMBER OF SIDE COOLERS/HEATERS  0

    FEED   STREAM  FEED
     NO      NO   PLATE
      1       1     7
      2      10     8

   PRODUCT STREAM  DRAW   DRAW
     NO      NO   PLATE   RATE
      1       2    14   0.64000
      2       3     0   ******

   CONDENSER TYPE-TOTL
   REBOILER TYPE -PART

   CONDENSER/DISTILLATE SPECIFICATIONS-
     DISTILLATE RATE0.64000 (D/F)

   REBOILER/BOTTOMS SPECIFICATIONS-
     REBOILER DUTY    6.00 KBTULB

   COLUMN PRESSURES & ESTIMATED TEMPERATURES
                  P(PSIA)   T(DEG F)
     CONDENSER    100.00    -11.00
     TOP PLATE    100.00
     REBOILER     100.00    236.00

   CONVERGENCE PARAMETERS
     NO OF ALLOWABLE CONSTANT MOLAL OVERFLOW ITERATIONS    0
     MAX ALLOWABLE ITERATIONS              20
     MAX DELTA T PER PLATE              20.000
     MAX FRACTIONAL LIQ CHANGE PER PLATE 0.400

   PLATE SPACING
     TOP SECTION   24.00 IN
     BOT SECTION   24.00 IN

   ESTIMATED LIQ RATE LEAVING TOP PLATE/CONDENSER 0.230 (L/F
   ESTIMATED BOTTOMS RATE 0.360 (B/F)
```

APPENDIX G

CHARACTERISTICS OF THE DISTILLATION

UNIT MODEL 8(B)

UNIT OPERATION NO 1 IS A DIST UNIT***

FEEDS>>>>>>>PRODUCTS>>>>>>>>>>>>>>>>

STREAM FLOW RATES ARE LB-MOLS

STREAM NO / NAME / COMPONENT	6	7	8
CH4	10.0867	8.4816	1.6051
C2H6	6.8303	1.4787	5.3516
C3H8	10.6636	0.3238	10.3398
IC4H10	2.7996	0.0183	2.7813
NC4H10	4.5195	0.0161	4.5034
IC5H12	1.6001	0.0012	1.5989
NC5H12	1.1680	0.0005	1.1675
NC8H18	1.4766	0.0000	1.4766
N2	0.0466	0.0454	0.0012
CO2	0.0664	0.0384	0.0280
TOTAL	39.2574	10.4039	28.8535
T,DEG F	-20.00	-37.80	86.54
P,PSIA	475.00	315.00	315.00
H,KBTU	-87.77	32.06	-5.79
S,KBTU/R	1.7234	0.4061	1.5564
MOL WEIGHT	41.6576	19.2120	49.7513
D,LB/FT3	33.7287	1.5420	32.2753
L/F(MOLAR)	1.00000	0.00000	1.00000

MINIMUM NUMBER OF STAGES = 1.61

MINIMUM REFLUX RATE = 0.92 LB-MOLS

NO STAGES IN COLUMN (INC REB)	REFLUX RATE LB-MOLS	CONDENSER DUTY 10E3KBTU	REBOILER DUTY 10E3KBTU	FEED TRAY REBOILER =TRAY 1
8.04	1.02	0.016	0.130	3.9
5.36	1.27	0.019	0.133	2.6
4.02	1.79	0.024	0.138	1.9
3.22	2.74	0.033	0.147	1.5
2.68	4.45	0.049	0.163	1.3

VITA

ANOUSHTAKIN ARMAN

Candidate for the Degree of

Master of Science

Thesis: ACCELERATION ALGORITHMS FOR PROCESS DESIGN SIMULATIONS

Major Field: Chemical Engineering

Biographical:

Personal Data: Born in Shiraz, Iran, February 20, 1960, The son of Badry Wilfong and Jalil Arman.

Education: Graduated from Cairo American College, Cairo Egypt, in May, 1977; received Bachelor of Science degree in Chemical Engineering from Oklahoma State University in May, 1985 and completed requirements for the Master of Science degree at Oklahoma State University in December, 1986.

Professional Experience: Lab assistant in the chemical and the metallurgical lab, Quality Control Department, Mercury Marine, Stillwater, Oklahoma, May, 1980 to June, 1982.