

A numerical method for finding the concentrations of chemicals in equilibrium

Leon Avery

Citation: [The Journal of Chemical Physics](#) **76**, 3242 (1982); doi: 10.1063/1.443317

View online: <http://dx.doi.org/10.1063/1.443317>

View Table of Contents: <http://scitation.aip.org/content/aip/journal/jcp/76/6?ver=pdfcov>

Published by the [AIP Publishing](#)

Articles you may be interested in

[Measurement method for finding gas adsorption equilibrium isotherms by employing a gas chromatograph and using its integrator in a new way](#)

Rev. Sci. Instrum. **76**, 044102 (2005); 10.1063/1.1889468

[Further simplification in the numerical determination of chemical equilibrium](#)

J. Chem. Phys. **84**, 6391 (1986); 10.1063/1.450732

[Numerical methods for solving the chemical mass action equilibrium problem](#)

J. Chem. Phys. **65**, 3085 (1976); 10.1063/1.433521

[Finding Hidden Symmetry by Numerical Methods](#)

J. Math. Phys. **11**, 3317 (1970); 10.1063/1.1665132

[Numerical Determination of Chemical Equilibrium and the Partitioning of Free Energy](#)

J. Chem. Phys. **46**, 4171 (1967); 10.1063/1.1840519



A numerical method for finding the concentrations of chemicals in equilibrium

Leon Avery

Department of Biochemistry, Stanford University Medical Center, Stanford, California 94305
(Received 28 September 1981; accepted 8 December 1981)

A new numerical method for solving the chemical equilibrium problem is described. It works by alternating projections of an estimate onto the set of all points representing systems at equilibrium and the set of all points consistent with conservation conditions. It is convenient to use and can handle large ill-conditioned systems quickly and accurately. It generalizes to nonideal solutions; however, it has only been tested for the ideal solution case as yet.

I. INTRODUCTION

One of the most important and common problems in chemistry and biochemistry is the calculation of the concentrations of chemicals at equilibrium. However, in all but the simplest cases the problem cannot be solved analytically. For this reason, several attempts have been made to find generally applicable numerical methods. White *et al.*,¹ Dantzig and DeHaven,² and Clasen³ described methods based on minimization of the free energy subject to conservation conditions. Unfortunately, these methods become very ill-conditioned when concentration ratios become large. Recently, Park⁴ described two new methods which, before starting the search for the equilibrium concentrations, attempt to manipulate the problem statement to a well-conditioned form. One of these, the prime method, is stable even when concentration ratios are large.

This paper describes a new method which I shall call the projection method. In tests, it has shown excellent convergence properties; i.e., it converged to within a relative accuracy of 10^{-6} in 3–4 iterations from estimates incorrect by many orders of magnitude. While the projection method is not inherently stable when concentration ratios become large, dynamic modification of the problem statement makes it so.

The method generalizes to nonideal solutions. In order to use it, it must be possible to compute activities from concentrations, concentrations from activities, and some approximation to the derivative matrix of the activities with respect to the concentrations.

The projection method is very different from Park's prime method and complementary to it. I have written a set of FORTRAN subroutines which employ techniques similar to the prime method to obtain an initial guess for the projection method, which is used to converge on the answer. So far the method has been tested only for the ideal solution case, but within this limitation it quickly and accurately solved all problems tested, including some large and very ill-conditioned systems.

II. THE DILUTE IDEAL SOLUTION

In all that follows, I will assume that the system under consideration is at constant temperature and pressure, so that the Gibbs free energy G is minimized at equilibrium. Furthermore, in this section, I will assume that

all reagents are solutes at such low concentrations that the volume doesn't change as the reactions proceed and activity corrections are negligible. Later, I will extend the discussion to systems in which these latter two assumptions cannot be made.

Let the number of reactions be R and the number of participating species N . The R reactions are assumed to be linearly independent. (If they are not, some of the dependent reactions can be eliminated, since they are irrelevant to the equilibrium. A FORTRAN subroutine has been written to do this and also to check that the equilibrium constants of the dependent reactions are consistent.) The reaction scheme is contained in the N by R matrix ν . ν_{ik} is the stoichiometry of the i th chemical species in the k th reaction, negative for reactants and positive for products. The natural logs of the equilibrium constants are contained in the R vector κ . The concentrations of the N species form an N vector \mathbf{c} , and the concentrations in the initial mix, before any of the reactions have advanced, is \mathbf{c}_0 . The natural logs of the concentrations form another N vector α . α may be thought of as relative chemical potential.

A useful property of ν follows from conservation of mass: for any nonzero R vector ξ , the N vector $\nu\xi$ has both positive and negative components. If this were not true, then $\nu\xi$ would represent a way of summing the reactions to get net creation of something from nothing.

There are two additional matrices that play important roles in the numerical methods to be described. The first, denoted by \mathbf{H} , is the N by N diagonal matrix with $h_{ii} = 1/c_i$, $i = 1$ to N . \mathbf{H} can also be defined as the matrix of derivatives of α with respect to \mathbf{c}

$$\mathbf{H} = (h_{ij}) = \left(\frac{\partial \alpha_i}{\partial c_j} \right) = \left(\frac{\partial \alpha}{\partial \mathbf{c}} \right). \quad (1)$$

Since the solution is ideal

$$\mu = \mu^0 + RT\alpha, \quad (2)$$

where μ is the vector of chemical potentials and μ^0 is independent of \mathbf{c} . (R in this expression is the gas constant, not the number of reactions. Since the former will always be multiplied by T and the latter never, no confusion should arise.) μ is defined by

$$\mu = \left(\frac{\partial G}{\partial \mathbf{n}} \right)^T,$$

where $n = Vc$ is the vector containing the total number of moles of each component present. Since V , the volume, is independent of c

$$H = \frac{V}{RT} \left(\frac{\partial^2 G}{\partial c^2} \right). \quad (3)$$

Except for a constant factor, H is the Hessian of G with respect to concentration.

The second important matrix is the R by R symmetric matrix

$$A = \nu^T H \nu. \quad (4)$$

A is proportional to the Hessian of G with respect to the reaction advancements. Since all the concentrations are positive, H is positive definite. This implies that A is positive definite, a reflection of the presence of a free energy minimum.

The equilibrium concentration vector is the intersection of two subspaces. One, which I will refer to as the constraint surface, is the set of all points representing systems at equilibrium. It is affine in α space and consists of those vectors satisfying

$$\nu^T \alpha = \kappa. \quad (5)$$

Equation (5) is obtained by taking the log of the familiar mass action expression.

The second subspace, called the feasible region, is the set of all points which can be reached from c_0 by advancing the reactions. It is an affine subspace of c space and is defined as the set of all c for which there exists an R vector ξ , so that

$$c = c_0 + \nu \xi. \quad (6)$$

ν , c_0 , and κ completely characterize the problem. There are equivalent problem statements, which can be recognized by the fact that they have identical constraint surfaces and feasible regions. One way to obtain an equivalent problem is to replace c_0 by any vector in the feasible region. Clearly, this will not change the constraint surface, since c_0 doesn't appear in its definition. To show that the feasible region is also unchanged, let $c'_0 = c_0 + \nu \xi_0$ be the vector replacing c_0 . Then, if c is any vector in the feasible region defined by c_0 , and c' any vector in the feasible region defined by c'_0 through Eq. (6)

$$\begin{aligned} c &= c_0 + \nu \xi & c' &= c'_0 + \nu \xi' \\ &= c_0 + \nu \xi_0 + \nu(\xi - \xi_0), & &= c_0 + \nu(\xi_0 + \xi'), \\ c &= c'_0 + \nu \xi', & c' &= c_0 + \nu \xi. \end{aligned}$$

Therefore, the feasible region as defined by c'_0 is identical to that defined by c_0 .

A second modification of the problem statement is the formation of linear combinations of the original reactions. If U is any nonsingular R by R matrix, then ν' and κ' defined by

$$\begin{aligned} \nu' &= \nu U^T, \\ \kappa' &= U \kappa, \end{aligned} \quad (7)$$

can be used to replace ν and κ in the original problem.

For if α and α' are any vectors in the constraint surface as defined by ν , κ and ν' , κ' , and c and c' any vectors in the feasible region defined by ν and ν' , respectively,

$$\begin{aligned} \nu^T \alpha &= \kappa, & (\nu')^T \alpha' &= \kappa', \\ U \nu^T \alpha &= U \kappa, & U^{-1} U \nu^T \alpha' &= U^{-1} U \kappa, \\ (\nu')^T \alpha &= \kappa', & \nu^T \alpha' &= \kappa, \\ c &= c_0 + \nu \xi, & c' &= c_0 + (\nu') \xi', \\ c &= c_0 + \nu U^T (U^{-1})^T \xi, & c' &= c_0 + \nu U^T \xi', \\ c &= c_0 + (\nu') \xi', & c' &= c_0 + \nu \xi. \end{aligned}$$

Thus, the constraint surface and the feasible region are both unchanged by replacement.

III. THE PROJECTION METHOD

The method is iterative. Letting c_i be the estimate of the equilibrium concentrations at the start of the i th iteration, the next estimate is found by the following four steps:

- (1) Compute α_i at c_i .
- (2) Project α_i onto the constraint surface to get α'_i .
- (3) Compute c'_i at α'_i .
- (4) Project c'_i onto the feasible region to get c_{i+1} .

Projection of a point onto a subspace can be defined as finding the member of the subspace closest to the point. The square of the distance between two points x and y is defined as

$$[(x - y)]^2 = (x - y)^T W (x - y), \quad (8)$$

where W is some weighting matrix, usually positive definite and symmetric.

Combining Eq. (5) and definition (8), α'_i of step (2) is the solution to the constrained minimization problem

$$\begin{aligned} \text{Minimize: } & (\alpha'_i - \alpha_i)^T M (\alpha'_i - \alpha_i), \\ \text{Subject to: } & \nu^T (\alpha'_i - \alpha_i) = \kappa - \nu^T \alpha_i, \end{aligned} \quad (9)$$

where M is a weighting matrix to be chosen later. In the terminology of Rao and Mitra,⁵ $(\alpha'_i - \alpha_i)$ is a minimum norm solution of Eq. (9). Since ν is full rank, the solution is unique and results in the following projection formula:

$$\alpha'_i = \alpha_i + M^{-1} (\nu^T M^{-1} \nu)^{-1} (\kappa - \nu^T \alpha_i). \quad (10)$$

The projection formula for step (4) is obtained by expressing c_{i+1} in the form of Eq. (6) and solving for ξ_{i+1} . ξ_{i+1} is the solution of the unconstrained minimization problem

$$\text{Minimize: } (\nu \xi_{i+1} + c_0 - c'_i)^T N (\nu \xi_{i+1} + c_0 - c'_i), \quad (11)$$

where N is the weighting matrix. Equation (11) defines ξ_{i+1} as the least-squares solution of

$$\nu \xi_{i+1} = c'_i - c_0. \quad (12)$$

Again, the fullrankness of ν allows unique solution:

$$c_{i+1} = c_0 - \nu (\nu^T N \nu)^{-1} \nu^T N (c_0 - c'_i). \quad (13)$$

It only remains to choose \mathbf{M} and \mathbf{N} . The simplest way to do this is to assume that the changes $\Delta\alpha$ and $\Delta\mathbf{c}$ in α and \mathbf{c} are small and approximately linearly related via

$$\Delta\alpha \sim \mathbf{H}(\Delta\mathbf{c}), \quad \Delta\mathbf{c} \sim \mathbf{H}^{-1}(\Delta\alpha). \quad (14)$$

It is then easily seen that $\mathbf{N} = \mathbf{M}^{-1} = \mathbf{H}$ is the correct choice. For, in step (2)

$$\begin{aligned} \mathbf{c}'_i &\sim \mathbf{c}_i + \mathbf{H}^{-1}\mathbf{H}\nu(\nu^T\mathbf{H}\nu)^{-1}(\kappa - \nu^T\alpha_i) \\ &\sim \mathbf{c}_0 + \nu[\xi_i + (\nu^T\mathbf{H}\nu)^{-1}(\kappa - \nu^T\alpha_i)], \end{aligned}$$

which is of the form of Eq. (6). Since \mathbf{c}'_i is also a member of the constraint surface, this means that it is the solution of the equilibrium problem in the linear approximation (14).

Similarly, in step (4)

$$\nu^T\alpha_{i+1} \sim \nu^T\alpha'_i + \nu^T\mathbf{H}[\mathbf{I} - \nu(\nu^T\mathbf{H}\nu)^{-1}\nu^T\mathbf{H}](\mathbf{c}_0 - \mathbf{c}'_i) = \kappa.$$

Thus, α_{i+1} , a member of the feasible region, is also approximately in the constraint surface.

In summary, the basic iteration is

- (1) $\alpha_i := \alpha(\mathbf{c}_i)$;
- (2) $\alpha'_i := \alpha_i + \mathbf{H}\nu\mathbf{A}^{-1}(\kappa - \nu^T\alpha_i)$;
- (3) $\mathbf{c}'_i := \mathbf{c}(\alpha'_i)$; and
- (4) $\mathbf{c}_{i+1} := \mathbf{c}_0 - \nu\mathbf{A}^{-1}\nu^T\mathbf{H}(\mathbf{c}_0 - \mathbf{c}'_i)$.

\mathbf{H} is evaluated at \mathbf{c}_i in step (2) and at \mathbf{c}'_i in step (4).

IV. NONIDEAL SOLUTIONS

The method described in Sec. III is applicable not only to dilute ideal solutions, but also to solutions in which activity coefficients differ significantly from unity and the volume is not constant. By generalizing the definitions of certain variables, the basic iteration remains almost unchanged.

First, consider the dilute nonideal solution, i.e., a solution dilute enough that the volume can be considered constant, but not so dilute that activity corrections are negligible. The major change is that Eq. (2) is taken as the definition of α , or equivalently, α is defined as the vector containing the natural logs of the activities. It is apparent that the constraint surface and the feasible region are still defined by Eqs. (5) and (6).

\mathbf{H} , defined by Eq. (1), is still proportional to the Hessian of G with respect to \mathbf{c} , and is therefore symmetric. Furthermore, \mathbf{A} is required to be positive definite for the projection method to be used. (Sufficiently close to a stable equilibrium \mathbf{A} will be positive definite, since it is the Hessian of G with respect to the reaction advancements.)

The use of \mathbf{H} as a weighting matrix affects the rapidity of convergence but not the accuracy of the solution obtained. Thus, approximations to \mathbf{H} can be used to assure positive definiteness or to reduce computation without jeopardizing the accuracy of the answer.

The removal of the final restriction, constancy of volume, is equally simple, although there is a conceptual problem to overcome. Concentration ceases to

be the relevant variable in the general problem and is replaced by \mathbf{n} , the total number of moles of each component present. Equation (6) is replaced by

$$\mathbf{n} = \mathbf{n}_0 + \nu\xi, \quad (15)$$

and \mathbf{H} is defined as

$$\mathbf{H} = \left(\frac{\partial\alpha}{\partial\mathbf{n}} \right) = \frac{1}{RT} \left(\frac{\partial^2 G}{\partial\mathbf{n}^2} \right). \quad (16)$$

\mathbf{n} , unlike \mathbf{c} , is an extensive variable. Since α is intensive, it is clear that the previously required one-to-one relationship between α and \mathbf{c} cannot be extended to α and \mathbf{n} . α determines \mathbf{n} only within a constant factor. This raises a problem in step (3) of the basic iteration: how should the constant be chosen? The surprising answer is that it doesn't matter; even 0 can be used. This is due to the following property of \mathbf{H} :

$$\mathbf{H}\mathbf{n} = 0, \quad (17)$$

where \mathbf{H} is evaluated at \mathbf{n} . Equation (17) is obvious from the fact that α is constant on a line leading from the origin through \mathbf{n} . It can also be derived from the Gibbs-Duhem equation. Since whatever \mathbf{n}'_i calculated in step (3) is annihilated by \mathbf{H} in step (4), it is clearly irrelevant what constant multiplies it. All the information about the composition of the system acquired in step (2) appears in \mathbf{H} in step (4).

Equation (17) implies that \mathbf{H} is not positive definite; at best, it is positive semidefinite with one zero eigenvalue. If the latter is the case, \mathbf{A} will still be positive definite, since the elements of \mathbf{n} are all positive, and any linear combination of the columns of ν has both positive and negative elements.

The basic iteration for the general case is therefore:

- (1) $\alpha_i := \alpha(\mathbf{n}_i)$;
- (2) $\alpha'_i := \alpha_i + \mathbf{H}\nu\mathbf{A}^{-1}(\kappa - \nu^T\alpha_i)$;
- (3) no action; and
- (4) $\mathbf{n}_{i+1} := (\mathbf{I} - \nu\mathbf{A}^{-1}\nu^T\mathbf{H})\mathbf{n}_0$.

V. NUMERICAL DIFFICULTIES

While the method described above gives excellent results if the problem is well-conditioned, it is unsatisfactory when the ratios of concentrations are large. Since concentration ratios very often are large, this difficulty cannot be ignored. The solution is to find an equivalent problem statement that is well-conditioned. There are two major numerical difficulties to be considered: (1) round-off error in the subtraction in step (4), and (2) singularity of the matrix \mathbf{A} in steps (2) and (4).

The first of these can be illustrated in the simplest of all chemical equilibria: the reaction $\mathbf{A} \rightleftharpoons \mathbf{B}$. Let

$$\begin{aligned} \nu &= [-1 \ 1]^T, \\ \kappa &= 4 \ln 10, \\ \mathbf{c}_0 &= [1 \ 1]^T, \\ \mathbf{c}_1 &= [1 \ 1]^T. \end{aligned}$$

Doing all calculations with a relative accuracy of 10^{-3} , the first iteration is

- (1) $\alpha_1 = [0 \ 0]^T$,
- (2) $\alpha'_1 = [-2 \ 2]^T \cdot \ln 10$,
- (3) $c'_1 = [0.01 \ 100]^T$,
- (4) $c_2 = [1 \ 1]^T - [1 \ -1]^T = [0 \ 2]^T$.

In step (4), all significance is lost in the calculation of the first component of c_2 because it is calculated as the difference of nearly equal numbers. This problem will clearly arise whenever an element of c_0 is much larger than the corresponding element of the equilibrium concentration vector. The solution is simple: since, as shown in Sec. II, c_0 can be replaced by any vector in the feasible region, replace it with one closer to the answer. The projection calculated in step (4) is in the feasible region and is the best choice based on available knowledge. Replacing c_0 with $[0 \ 2]^T$ in the example and repeating step (4)

$$(4) \ c_2 = [0 \ 2]^T - [-2 \times 10^{-4} \ 2 \times 10^{-4}]^T = [2 \times 10^{-4} \ 2]^T,$$

which is the correct answer to the required precision.

The generalization of the example is simple: when doing the projection onto the feasible region, check whether any component is much less than the corresponding element of c_0 . If so, replace c_0 with the projection vector just calculated and repeat the projection. This cycle may have to be repeated if concentration ratios are very large, but since each replacement reduces the error by a factor corresponding to the relative precision of the calculations ($<10^{-6}$ on most computers), an accurate answer is quickly obtained. Furthermore, the work need not be repeated in subsequent iterations.

The second numerical difficulty is more complex. Let

$$\nu = \begin{bmatrix} -1 & 0 \\ 1 & -1 \\ 0 & 1 \end{bmatrix},$$

$$\kappa = [-6 \ 6]^T \cdot \ln 10,$$

$$c_0 = [1 \ 1 \ 1]^T,$$

$$c_1 = [1 \ 1 \ 1]^T.$$

The first iteration goes

- (1) $\alpha_1 = [0 \ 0 \ 0]^T$,
- (2) $\alpha'_1 = [2 \ -4 \ 2]^T \cdot \ln 10$,
- (3) $c'_1 = [100 \ 10^{-4} \ 100]^T$.

Now, when step (4) is attempted,

$$A = \nu^T H \nu = \begin{bmatrix} 10^4 & -10^4 \\ -10^4 & 10^4 \end{bmatrix} + \begin{bmatrix} 0.01 & 0 \\ 0 & 0.01 \end{bmatrix} \sim \begin{bmatrix} 10^4 & -10^4 \\ -10^4 & 10^4 \end{bmatrix}.$$

The matrix is singular, and the projection formulas can't be used.

However, an equivalent problem is easily found in which this difficulty does not arise. Let

$$U = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$$

and transform ν and κ according to Eqs. (7)

$$\nu' = \begin{bmatrix} -1 & -1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \kappa' = [-6 \ 0]^T.$$

Now

$$A' = (\nu')^T H (\nu') = \begin{bmatrix} 10^4 & 0.01 \\ 0.01 & 0.02 \end{bmatrix}.$$

This matrix is distinctly nonsingular and c_2 is easily calculated.

The essential difference between the first and the second form of the problem above is that A' is much closer to being a diagonal matrix than A . Thus, the small contributions resulting from the larger concentrations are not obscured by the large contribution of the second component, which has all been packed into the first diagonal element. This diagonalization of A is the property used to generalize the example

$$A' = (\nu')^T H (\nu') = U A U^T. \quad (18)$$

There are many ways to choose U so that A' is diagonal; the best is that which can be found and applied with the least possible additional effort. The choice can be made most efficiently if it is coupled with the equation solving process.

In both projection steps, it is necessary to solve equations

$$A x = y. \quad (19)$$

In step (2), $y = \kappa - \nu^T \alpha_i$ and x is a vector of Lagrange multipliers; in step (4) $y = \nu^T H (c_0 - c'_i)$ and x is the advancement vector. Although $A^{-1}y$ has been written in the projection formulas for it, x is not computed by inverting A and applying the inverse to y ; this would be inefficient. Rather, the equations are solved by the usual method of reducing A to triangular form, then solving the triangular system of equations directly.

Since A is known to be positive definite, a particularly efficient reduction to triangular form is available: the Cholesky decomposition. A is expressed as the product of an upper triangular matrix L and its transpose

$$A = L^T L. \quad (20)$$

It is simple to compute L given A .⁶ x in Eq. (19) is found by direct solution of the two successive triangular systems of equations.

As in the example, A may become positive semi-

definite due to large concentration ratios, in which case the Cholesky decomposition breaks down. However, an extension of the Cholesky decomposition is useful in this case

$$A = P^T L^T L P. \quad (21)$$

If r is the rank of A , L is an upper triangular r by R matrix. (In this context "upper triangular" means that $i > j$ implies $l_{ij} = 0$ for $i = 2$ to r , $j = 1$ to $r - 1$.) P is a permutation matrix necessary to switch the first r rows of $L^T L$, which are independent, into their proper positions in A . L is computed exactly as in an ordinary Cholesky decomposition, except that at every stage care is taken to switch a positive pivot into position. The switches are kept track of and comprise P . When there are no remaining positive diagonal elements, the decomposition is complete.

If A is positive definite all pivots will be positive, the decomposition will proceed all the way through A , and Eq. (19) can be solved for x . But if $r < R$, A is singular and some transform U must be found which, when applied to v , will result in a more diagonal A' . Such a transform can be obtained directly from the incomplete Cholesky decomposition. Let

$$U = \begin{bmatrix} L^T & 0 \\ I & I \end{bmatrix}^{-1} P. \quad (22)$$

Then, using Eq. (21) and the orthogonality of permutation matrices

$$U A U^T = \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix},$$

a diagonal matrix.

U is never explicitly computed. Only L and P need ever be present in memory, and the latter only in schematic form. Since solution of a triangular system of equations is just as fast as multiplication by a triangular matrix, explicit inversion would be a waste of time. Any row of U may be multiplied by a constant in order to keep v and κ order 1; this sort of normalization may be convenient when doing calculations by hand.

Thus, the detection of singularity in A and its correction are direct results of the equation solving process. A is computed and factored according to Eq. (21). If it was singular, v and κ are replaced and the computation and factoring of A are repeated. This cycle, like that of replacing c_0 , may have to be repeated, but it results quickly in an accurate answer and the replacements, once computed, can be used in subsequent iterations. Note that y is not needed for any of this. y is computed and x solved for only after a nonsingular A has been computed and factored. Moreover, if the projection in step (4) needs to be repeated because c_0 was replaced, it is not necessary to compute or factor A again. The calculation and Cholesky decomposition of A take $O(R^3/3 + NR^2)$ multiplications and additions, but all the rest of the work in the step only $O(R^2 + 4NR)$, so the cost of repetition of step (4) becomes negligible for large systems.

VI. INITIAL GUESSES

Like all iterative methods, the projection method requires an initial guess. Unlike most iterative methods, it is very forgiving with regard to the initial guess and often converges quickly from a very bad one. In several complex systems, I have obtained convergence in three-four iterations from guesses that were incorrect by as much as 30 orders of magnitude. However, there are systems in which very poor initial guesses result in nonpositive concentrations in step (4), and in these cases some thought must be given to the problem.

A vertex of the feasible region is used for the initial guess. Although neither of steps (1) and (2) can be done if any of the concentrations is zero, α'_i approaches a finite limit as some of the elements of c_i go to zero. Thus, it is possible to use a vertex to start the iteration, although a special routine was necessary to handle the first projection. All the zero concentrations are given equal weight in this projection. (As long as the vertex is not degenerate, the usual case, the weighting is irrelevant.)

The vertex is chosen by a programming search similar to that used by Park⁴ in his prime method. My search routine differs from his in the following details: (1) linear programming techniques are used instead of integer programming; (2) the "goodness" of a vertex is measured not by the deviation from equilibrium at the midpoints of the edges, but rather by estimating the distance of the equilibrium point of each reaction from the vertex; and (3) the search is not stopped when a vertex better than all its neighbors is found, but only when some vertex is encountered for the second time. The best of all previous vertices is then used as the initial guess. These differences are probably not very important; the two search methods accomplish the same basic function. In addition to finding an initial estimate, the programming search provides a well-conditioned problem statement to start off with.

VII. RESULTS AND DISCUSSION

FORTAN routines have been written to solve the dilute ideal solution problem using the projection method. Routines to find initial guesses by a vertex search have also been written. Together, these programs solved quickly and accurately all systems tested. However, I have not yet tested the projection method on any of the nonideal systems discussed in Sec. IV.

Using as initial estimate a vector containing all ones, the examples given by White *et al.*¹ and Park⁴ were solved by the projection method. In all three cases, the iteration converged in three iterations. Note that the initial estimate was incorrect by a factor of $> 10^3$ for one of the components in White *et al.* example, and by $> 10^6$ in Parks's first example. Better performance would probably have been obtained in the latter by starting with a vertex search.

The method has also been tested on much more complicated and ill-conditioned examples. Table I shows v , κ , and c_0 for a very ill-conditioned 7 by 15 system.

TABLE I. This table describes one of the problems used to test the projection method. The 7 by 15 matrix is the stoichiometry for the reaction system, products positive and reactants negative. (For instance, the second reaction is $C + D + E \rightleftharpoons 2A + B$.) The row at the bottom gives the base 10 log of the equilibrium constant for each reaction. (The equilibrium constant of reaction 2 is 10^{48} .) The column at the right gives the initial concentration of each reagent. (The initial concentration of B is 1.)

Reagent:	Reaction number							c_0
	1	2	3	4	5	6	7	
A	2	2	2	2	2	2	2	1
B	-1	1	1	1	1	1	1	1
C	-1	-1	1	1	1	1	1	1
D	0	-1	-1	1	1	1	1	1
E	0	-1	-1	-1	1	1	1	1
F	0	0	-1	-1	-1	1	1	1
G	0	0	-1	-1	-1	-1	1	1
H	0	0	0	-1	-1	-1	-1	0.25
I	0	0	0	-1	-1	-1	-1	0.25
J	0	0	0	0	-1	-1	-1	0.25
K	0	0	0	0	-1	-1	-1	0.25
L	0	0	0	0	0	-1	-1	0.25
M	0	0	0	0	0	-1	-1	0.25
N	0	0	0	0	0	0	-1	0.75
O	0	0	0	0	0	0	-1	0.75
$\kappa/\ln 10$:	18	48	54	18	-30	-90	-90	

The solution of this system by the programs mentioned above is described in Table II. An extended 63 by 127 version of the same system was the most complicated problem tested. (The first 62 elements of the equilibrium concentration vector formed a geometric progression from 1 to 10^{-30} ; the remaining elements were 1's.) 2.9 s were required to find the initial estimate and 2.1 s to converge to the desired precision of 10^{-4} . (The program was run on an IBM 370/168 processor.)

The aim of this work was to develop automatic methods for the simulation of chemical equilibria. For the ideal solution case, the methods described here seem to suffice. The projection method has proven capable of handling large and very ill-conditioned systems of reactions when complemented by a programming search similar to that employed by Park in his prime method. The general routines are convenient to use and require very little knowledge of computer programming on the part of the user. Therefore, they should allow convenient simulation of reaction systems for a variety of purposes.

There is some room for improvement in step (4) of the basic iteration: projection onto the feasible region. As previously mentioned, it may result in negative concentrations if the initial estimates are poor. This problem could be overcome by redefining the feasible region as the convex subset of the space defined by Eq. (6), which is bounded by the inequality constraints $c \geq 0$. Step (4) would consist of minimizing Eq. (11) subject to these constraints; a quadratic programming problem, which can be solved by a simplex method.⁷

For nonideal solutions, it is more difficult to apply these methods. Step (3) requires that concentrations be computed from activities, which means the inversion

of an N -dimensional function α of the N vector c . For weakly nonideal solutions where activity coefficients vary only slowly with c , this is not really hard: c' can be estimated using $c' = c \exp(\alpha' - \alpha)$. The estimate could be refined by Newton's method, slow variation of activity coefficients ensuring rapid convergence, but it is not necessary. Newton's method is essentially built into the projection method by using H . But in strongly nonideal solutions where activity coefficients vary rapidly, this is likely to be inadequate. Simple preliminary cal-

TABLE II. This table shows the solution of the problem in Table I by the projection method. Iteration 0 is the initial estimate of the equilibrium concentration vector obtained by a programming search of the vertices of the feasible region. Iterations 1 and 2 are the concentration vectors obtained after one and two iterations of the projection method. After the second iteration, the relative error is $\leq 3 \times 10^{-5}$ for all species.

Reagent	Iteration		
	0	1	2
A	0	1.010691	1
B	0	9.98668E-07	1E-06
C	0	1.01579E-12	1E-12
D	0	9.98692E-19	1.00002E-18
E	0	9.98696E-25	1.00003E-24
F	0	9.98700E-31	1.00003E-30
G	0	0.999109	1
H	1.125	0.998663	0.999999
I	1.125	0.998663	0.999999
J	1.125	0.998663	0.999999
K	1.125	0.998663	0.999999
L	1.125	0.998664	1
M	1.125	0.998664	1
N	1.625	0.999109	1
O	1.625	0.999109	1

culations modeling the aqueous phosphate–phosphoric acid equilibrium generally bore out these expectations.

ACKNOWLEDGMENTS

I wish to thank Dr. T. Jovin and Dr. R. Clegg for their encouragement in all stages of this work.

¹W. B. White, S. M. Johnson, and G. B. Dantzig, *J. Chem. Phys.* **28**, 751 (1958).

²G. B. Dantzig and J. C. DeHaven, *J. Chem. Phys.* **36**, 2620 (1961).

³R. J. Clasen, *The Numerical Solution of the Chemical Equilibrium Problem*, Rand Corporation Memorandum RM-4345-PR (1965).

⁴D. J. M. Park, *J. Chem. Phys.* **65**, 3085 (1976).

⁵C. R. Rao and S. K. Mitra, *Generalized Inverse of Matrices and Its Applications* (Wiley, New York, 1971).

⁶Y. Bard, *Nonlinear Parameter Estimation* (Academic, New York, 1971), pp. 307–309.

⁷Johannes Cornelius Gerardus Boot, *Quadratic Programming* (North Holland, Amsterdam, 1964).