Digital Computer Approaches to the Power Load Flow Problem

RODNEY J. BROWN

Bonneville Power Administration
United States Department of the Interior
Portland 8, Oregon

RECENT INTENSIVE and widespread use of high-speed digital computational equipment has made areas of mathematics fruitful for investigation that previously were considered unattractive due to the extreme tedium of mass calculation by hand methods. This paper will use as a case history the pioneering efforts of the Bonneville Power Administration (BPA) in the development of iterative numerical processes for solving large-scale networks for electrical power flow to illustrate some of the areas of mathematical interest.

The Administration has had over five years of actual experience in using digital computers to solve power system problems, and staff interest in the subject extends back much further. The BPA acquired a Bendix G-15 computer in June, 1955, and an IBM 650 computer in April, 1956. In addition, considerable use is being made of the IBM 704 at Tennessee Valley Authority and the IBM 709 at Richland, Washington.

The Power Load Flow—as an Engineering Problem

The problem confronting the BPA engineering staff was the necessity of solving the complex flow of power on the Pacific Northwest Power Pool Grid with high enough accuracy to compare system power losses from case to case. The analog methods previously used did not provide this order of accuracy. This necessitated an entirely new approach to an old problem.

The solution of electric power load flow problems is one of the most important tasks in system analysis. Consequently, considerable effort has been devoted to developing methods for solving load flow problems on a digital computer.

At least three different methods of solution have been developed and employed with varying degrees of success. Each method has possibilities for further improvements, and each offers advantages over the conventional network analyzer approach.

As early as 1947, successes were obtained by Dunstan with a scheme known as the "loop and track method." This requires an inversion of a complex matrix of order equal to the number of independent loops in the system and a modification of this inverse for every branch change involved in subsequent problems on the same system. This calculation becomes unwieldy as the system becomes large and imposes a size limitation less than that required for most practical problems. Data preparation is not as simple as in the other methods. A set of independent loops and a connection pattern in the form of tracks must be selected and carefully coded. This method is presently best suited for problems involving small systems or few loops. For these smaller problems the method is faster than any other.

More recently (Ward and Hale, 1956; Brown and Tinney, 1957), attention has been given to a scheme called the nodal iterative method in solving load flow problems. In this method the problem is represented by the second-degree nodal power equations. These equations are solved for the node or bus voltages by various iterative schemes, including the Gauss-Seidel process with overcorrections and extrapolation. Output quantities are computed from the bus voltages and circuit parameters. Any pair of the following terminal conditions can be maintained at any bus: real power, reactive power, voltage magnitude, voltage angle, volt-amperes. Transformer tap settings can be represented rigorously. Any practical degree of precision can be obtained, and it appears that problems involving any size or configuration of system can be solved. Systems as large as 250 busses and 400 branches have been handled using only a medium-sized computer with temporary card storage to augment the limited drum memory.

The procedure for performing a digital load flow analysis by this method is typically as follows. A system diagram is prepared in the traditional manner, as for a network analyzer study. In addition, every node or bus must be numbered. Special data forms are then filled out. The data entered on these forms are: the bus names and numbers with the desired power, reactive or voltage magnitude for each bus; the line names and numbers, which are a composite of the terminating bus pair names and numbers, and the line impedances and pi equivalent shunt admittances; the transformer names and numbers, which are assigned in the same manner as line numbers; and the tap settings and impedances of the transformers. These data are punched on cards. The data cards are read into the computer, where they are converted into a set of appropriate equations, and the load flow solution for the given conditions starts immediately. After an indefinite period, depending on system size, configuration, and desired precision, a bus voltage solution is reached. At this time the machine starts computing and punching cards containing the desired output quantities. With machines of limited internal storage, it may be necessary to re-enter the input data at this time. The solution gives the voltages of every bus in polar form, the real and reactive power flows and losses in every line and transformer, the reactive that must be supplied at every bus under voltage control, the total system losses, and, optionally, several other useful answers. These data, which are punched on cards, can be listed in any convenient tabular form, or with additional processing may be printed directly on a network diagram. The program can be arranged to sequence automatically from one case to the next by merely reading in data cards for the changes.

Using an IBM 650, a 250-bus problem requires about two hours for analyzer precision and up to five hours for precision suitable for loss studies. In the manner in which most studies are performed, it is possible to start each new solution with the bus voltage solution of a similar case. This usually reduces the solution time as compared with taking a completely arbitrary start. Some loosely coupled systems are apparently poorly conditioned for this type of solution and require much more time than others for convergence to acceptable precision. Improvements in the iteration technique may overcome this disadvantage. Although the running time and cost are large when high precision is required for loss studies, the solution obtained in this case is impossible without the computer. It is likely that the nodal method will be further improved in speed, flexibility, and convenience. This discussion is in terms of an IBM 650 computer. Most of the BPA iterations are now performed on IBM 704 and IBM 709 with 75 times the speed.

Another quite different type of load flow solution by Shipley and Hochdorf (1956) has been used as an element in an involved program for solving an economic dispatch problem. This method uses the open-circuit impedance matrix of the network. Bus currents and voltages are improved by an iterative scheme to meet the given bus power and reactive schedule. The development of the impedance matrix needed for this method is an extensive problem in itself. The matrix contains a number of complex elements equal to the square of the number of busses. The number of iterations for convergence to acceptable precision is apparently very few compared with the nodal method, but the total amount of computation is at least comparable for practical-sized problems. This method is suitable only on large computers and has not been as extensively used as the nodal method.

The Power Load Flow—as an Applied Problem in Numerical Analysis

Only the nodal iterative scheme of solution will be considered. This scheme requires the approximate solution of a set of simultaneous second-degree equations with complex variables and coefficients of the form:

$$(1) \quad E_i \sum_{j=1}^{n} Y_{ij}E_j = P_i + jQ_i$$

$$i = 1, 2, \ldots, n.$$

The E's are in general the unknowns (sometimes the Q_i and the angle of the E_i).

Thinking of the Y's in matrix array with a 250-node problem, we have a problem with 250 equations in 250 complex unknowns, nonlinear, to be solved. At first blush such a problem appears wholly out of range for our 2,000-word computer. We have factors that can be used to our advantage, however, for the array is very sparse and is symmetric. The system is known to be well behaved in at least the electrical sense. Linear approximations in the neighborhood of solutions are adequate.

The solving process adopted at BPA is somewhat as follows. Estimates of E values are made. The equations are then solved for E_i holding the E_j constant by a combination of iterative schemes.

1. Successive overcorrected displacements

   (a) Solve left hand of equation (1) obtaining value of $P + jQ$ by using estimated or improved E's. Obtain difference between computed values and the scheduled $P + jQ$ yielding residuals p and q.
   (b) Multiply p by 1.7 and q by 1.3 and substitute in a linear expression for $\triangle E$ and add to E obtaining overcorrected E_i. Put new E_i in storage and calculate new $E_i + \triangle$.

2. Simultaneous undercorrected displacements

   These are used after a suitable number of successive displacements have been made.
   (a) Same as successive.
   (b) Same as successive except undercorrected p by 0.5 and q by 0.3.
   (c) Store corrections to E_i until all nodes have been computed and then add all corrections at one time at the end of an iteration.
   (d) Return after one iteration to successive method.

3. Extrapolation with successive displacements

   There are several methods that have proved effective (Brown and Tinney, 1957). One method depends upon the ratio of the sequence of corrections of each variable approximating a series. (N.B. This uses the eigenvalue (Van Ness, 1959).)

(2) (E_i) extrapolated $= (E_i)_{n+1} \frac{r}{r-1} (\triangle E_i)_n$

where r is the ratio of the geometric series

$$r = \frac{(E_i)_{n+1}}{(E_i)_n}$$

When ratio is 0.9 or less, this method produces a sufficiently precise solution to the problem as soon as successive convergence pattern is monotonic. Where the ratio value is 0.95 or greater, the method is applied several times with intervening successive iterations.

The actual arithmetic used by BPA in its iterative program for the IBM 704-709 as developed by William Tinney incorporates many refinements looking toward the minimization of computer operations. These refinements are derived by an actual manipulation and study of the number and nature of cross multiplications required in the development of residuals p and q. These are all that are required in the linear adjustment between sequential iterations.

Some Mathematical Conjectures

Most of the parameters and combinations of methods used in our iterative solutions have come from empirical reasoning derived from tedious observations of masses of numerical data. Many of the key clues leading to successful methods have come from our knowledge of physical electrical systems and their practical operation. Unfortunately, very little has come from the mathematical literature. This fact is our basic reason for this discussion.

Many of the things we have learned by tedious and devious means were found to be known and available although we did not discover them until years later. For example, the extrapolation ratio we found by experiment is developed by Van Ness (1959) and shown to be the largest eigenvalue of the system. The overcorrection of 1.7 used, which was found by multiple experiments, has been shown to be theoretically expected. We feel that these lags in communication between those who explore the mathematical boundaries and those who attempt difficult problem solution are due to a definite "gray zone" between the two groups. We feel this can and should be eliminated. We have much to be gained by intimate and continuing cross communication. Traditionally, mathematics has been theoretical only, with little or no laboratory verification. We now have in computing equipment enormous logical engines capable of running out previously impossible displays of combinations. Observation of these patterns will lead the mind to new conjectures beyond those possible without these laboratory "props." We know we need help from those who have explored the boundaries, and we feel sure that they too can derive consolation by observing our efforts.

We have developed a method that solves a problem with many unknowns economically on a medium-sized computer and in minutes on a large one. Our system, like most physical systems viewed in their proper duality, is quite sparse (Ullah, 1959). Therefore, we are sure that what we have learned in the electrical power business could be useful elsewhere if an adequate means of communication could be developed.

Classical methods will continue to extend our boundaries of knowledge, but they will be aided by short-cut tricks in the problem-solving vineyard, which may lead to further conjectures to extend again the boundaries of our knowledge.

Literature Cited

Brown, R. J. and W. F. Tinney. 1957. Digital solutions for large power networks. AIEE Transactions, No. 57-147, January, 1957. Power apparatus and systems, June, 1957, pp. 347-351.

Dunstan, L. A. 1947. Machine computation of power network performance. AIEE Transactions, Vol. 66, Section P787.

Shipley, R. B. and M. Hochdorf. 1956. Exact economic dispatch-digital computer solution. AIEE Transactions, Vol. 75, Pt. III, December, 1956, pp. 1147-1152.

Ullah, A. 1959. Simplified formulas for the digital computation of power flow in an electrical network and some new aspects of convergence. Ph.D. Thesis, Texas A&M College.

Van Ness, J. E. 1959. Northwestern University. Convergence of iterative load flow studies. AIEE 59-1092. Fall General Meeting. Chicago, Illinois.

Ward, J. B. and H. W. Hale. 1956. Digital computer solution of power flow problems. AIEE Transactions, Vol. 75, Pt. III, June, 1956, pp. 398-404.