Systematic Scaling for Digital Differential Analyzers*

ARTHUR GILL†

***Summary*—The usefulness of large-capacity digital differential analyzers (DDA's) is severely hampered by the complexity of the scaling process. The scales needed for programming a DDA have to be compatible with the so-called "equilibrium," "topological," and "boundary" constraints, imposed by the construction of the analyzer and the nature of the problem at hand. Simultaneous trial-and-error satisfaction of all these constraints, to achieve optimal range and accuracy of computation, is practically impossible for any problem involving more than a few integrators. The paper shows how the scaling constraints can be organized in a matrix form, and how optimal scales can be produced in a systematic manner. The proposed scheme, which can be programmed for automatic execution, is adaptable for DDA's operating in conjunction with general-purpose digital computers.**

Introduction

FROM a functional standpoint, a digital differential analyzer (DDA) consists of packages, each containing an integrator and an associated constant multiplier. The integrator receives incremental inputs of two different types, called the dy and dx inputs. The dy inputs are accumulated in a register to form the integrand y. The increments dx of the variable of integration x control the addition (or subtraction) of y into another register, called the r register. Overflows of r are increments of the integral of y with respect to x and can be accumulated in another integrator. The integral of y with respect to x is called z, and the increments of z are called dz. The dz outputs of each integrator control the addition (or subtraction) of a constant k into a register called the k_r register. Overflows of k_r are called kdz outputs of the integrator, and can serve as inputs (dx, dy or both) to other integrators. The kdz outputs represent increments of the integral of ky with respect to x. An integrator may have only one dx input, but as many dy inputs as permitted by the capacity of the dy accumulators. Fig. 1 is a schematic representation of an integrating package in a DDA.

Integrating packages of the type described above can be interconnected to provide digital solutions to differential or algebraic equations—linear or nonlinear, single or simultaneous. Fig. 2 shows, as an example, an interconnection of integrators to provide the solution to

$$\frac{d^2y}{dt^2} + \frac{dy}{dt} - y^2 - \sin y = 0.$$

The solution is registered in integrator no. 3 and can be typed out periodically during computation.

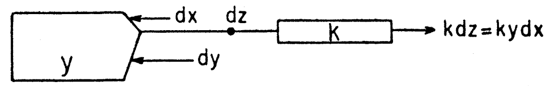

Fig. 1—An integrating package.

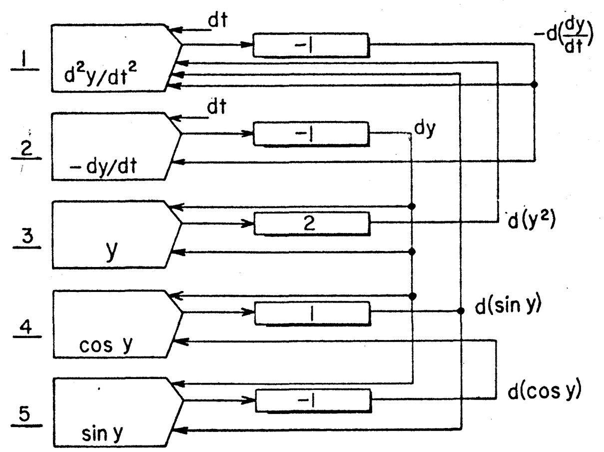

Fig. 2—Integrator network for

$$\frac{d^2y}{dt^2} + \frac{dy}{dt} - y^2 - \sin y = 0.$$

Further details concerning the construction and programming of DDA's can be found in the references.[1-3]

DDA Scaling

In a digital differential analyzer the quantities y, k, dx, dy, and dz are manipulated under the fixed-point system. The position of each quantity with respect to the binary point in the register is dictated by a "scale" associated with that quantity. Specifically, this scale equals the power of 2 by which the register quantity has to be multiplied in order to yield the true value. In the following discussion α_i, β_i, γ_i, δ_i, and ϵ_i will denote the scales associated with the y, k, dx, dy and kdz quantities, respectively, in the ith integrating package.

If there are N packages in a given problem, $5N+1$ scales have to be specified by the programmer, *i.e.*, 5 scales for each participating package, plus a scale denoted ϵ_0 for the independent variable. These scales cannot be specified independently. First, for each integrating package the following "equilibrium constraint" has to be satisfied:

$$\alpha_i + \beta_i + \gamma_i = \epsilon_i \qquad i = 1, 2, \cdots, N. \tag{1}$$

* Manuscript received by the PGEC, June 8, 1959; revised manuscript received, August 19, 1959. The work described in this paper was done for the Bendix Computer Div. of the Bendix Aviation Corp., Los Angeles, Calif.

† Dept. of Electrical Engineering, University of California, Berkeley, Calif.

[1] G. F. Forbes, "Digital Differential Analyzers," G. F. Forbes Publication, Pacoima, Calif ; 1956.

[2] M. Palevsky, "The design of the Bendix digital differential analyzer," Proc. IRE, vol. 41, pp. 1352–1356; October, 1953.

[3] "Programming Manual for the DA-1—Digital Differential Analyzer Accessory for the Bendix G-15D Computer," Bendix Computer Div. of Bendix Aviation Corp., Los Angeles, Calif.; 1957.

Second, the interconnection diagram implies "topological constraints" of three types: 1) Since every dx input is a kdz output, every γ_i has to be equal to some $\epsilon_j (i \neq j)$. 2) Since every dy input is a kdz output, every δ_i has to be equal to some ϵ_j. 3) If the kdz outputs of integrators $a, b, \cdots, g$ all serve as dy inputs to the same integrator, then $\epsilon_a = \epsilon_b = \cdots = \epsilon_g$.

In order to realize the highest computational accuracy, it is necessary to choose α and β scales such that the corresponding registers will accumulate the largest possible number of significant digits without overflowing during the problem run. Scales for which the above condition is realized will be called "optimal scales." Letting $\hat{y}_i$ be the maximum value that the ith integrand assumes in a given problem, the optimal α and β scales are then given by:

$$\alpha_i = \{\log_2 \hat{y}_i\} \tag{2}$$

$$\beta_i = \{\log_2 k\} \tag{3}$$

where $\{a\}$ denotes the smallest integer which equals or exceeds a.

Determining a set of scales compatible with the equilibrium and topological constraints, and at the same time realizing optimal operation, is seen to be quite involved when the problem at hand requires more than 10 or 15 integrators. Establishing the scales by trial-and-error methods is tedious at best, and often unfeasible. In the following sections a procedure will be described by which scales can be produced in a systematic manner, possibly with the aid of a digital computer.

MATRIX FORMULATION OF THE SCALING CONSTRAINTS

Considering the example described by Fig. 2, the equilibrium constraints can be written as

$$\alpha_i + \beta_i + \gamma_i = \epsilon_i \qquad i = 1, 2, \cdots, 5. \tag{4}$$

The topological constraints of type 1) are

$$\gamma_1 = \epsilon_0, \quad \gamma_2 = \epsilon_0, \quad \gamma_3 = \epsilon_2, \quad \gamma_4 = \epsilon_2, \quad \gamma_5 = \epsilon_2; \tag{5}$$

of type 2) they are

$$\delta_1 = \epsilon_1, \quad \delta_2 = \epsilon_1, \quad \delta_3 = \epsilon_2, \quad \delta_4 = \epsilon_5, \quad \delta_5 = \epsilon_4; \tag{6}$$

and of type 3) they are

$$\epsilon_1 = \epsilon_3 = \epsilon_4. \tag{7}$$

Substituting (7) in (5) and then eliminating the γ's from (4), yields

$$\left.\begin{aligned} \alpha_1 + \beta_1 + \epsilon_0 - \epsilon_1 &= 0 \\ \alpha_2 + \beta_2 + \epsilon_0 - \epsilon_2 &= 0 \\ \alpha_3 + \beta_3 + \epsilon_2 - \epsilon_1 &= 0 \\ \alpha_4 + \beta_4 + \epsilon_2 - \epsilon_1 &= 0 \\ \alpha_5 + \beta_5 + \epsilon_2 - \epsilon_5 &= 0 \end{aligned}\right\}. \tag{8}$$

The above steps can be carried out in any given problem to yield a set of equations of the form

$$\alpha_i + \beta_i + \epsilon_j - \epsilon_k = 0. \tag{9}$$

The number $M+1$ of different ϵ's appearing in (9) is smaller than $N+1$ for all problems in which there is at least one integrator having more than a single dy input. Since this is the case in all problems involving addition (or subtraction), and hence in all nontrivial problems, it will be invariably assumed that $M<N$. By noting that ϵ_0 always has to appear in (9), and that the numbering of the integrating packages can be chosen arbitrarily, it can be assumed without loss of generality that the ϵ's appearing in (9) are $\epsilon_0, \epsilon_1, \cdots, \epsilon_M$. Using this assumption, the coefficient matrix for (9) can be written as shown in (10).

$$\begin{array}{c} \\ 1 \\ 2 \\ \cdot \\ \cdot \\ N \end{array} \begin{array}{c} \begin{array}{cccccccccccccccc} \alpha_1 & \alpha_2 & \cdot & \cdot & \alpha_N & \beta_1 & \beta_2 & \cdot & \cdot & \beta_N & \epsilon_0 & \epsilon_1 & \cdot & \cdot & \epsilon_M \end{array} \\ \left[\begin{array}{cccccccccccccccc} 1 & 0 & \cdot & \cdot & 0 & 1 & 0 & \cdot & \cdot & 0 & & & & & \\ 0 & 1 & \cdot & \cdot & 0 & 0 & 1 & \cdot & \cdot & 0 & & & & & \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & & & & & \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & & & & & \\ 0 & 0 & \cdot & \cdot & 1 & 0 & 0 & \cdot & \cdot & 1 & & & & & \end{array}\right] \end{array}. \tag{10}$$

$$\underbrace{\qquad\qquad}_{\text{Identity matrix}} \quad \underbrace{\qquad\qquad}_{\text{Identity matrix}} \quad \underbrace{\qquad\qquad}_{\substack{\text{Each row contains 2} \\ \text{unities and } M-1 \text{ zeros}}}$$

Matrix (10) shows that, out of the $5N+1$ scales to be determined, only $N+M+1$ can be independently specified. The independently specifiable scales correspond to those columns in (10) which, when deleted, leave a nonsingular matrix. It is also evident that a nonsingular matrix, and hence unique values for all scales, can always be produced by leaving either all the α's or all the β's (or a mixed set of N α's and β's) unspecified. This scheme, however, is of little value, since it is always desirable to preserve the freedom of specifying as many α's and β's as possible, so that optimal scales can be guaranteed at the outset. It is also imperative to be able to specify ϵ_0 independently, since this scale constitutes the only means by which the speed of computation can be directly controlled. Thus, the task at hand is to find in (10) a nonsingular $N \times N$ matrix, which contains the least number of α and β columns and which does not contain the ϵ_0 column.

THE ε MATRIX

The best one can do to establish optimal scaling is to find in (10) a nonsingular matrix which contains all the ϵ columns exclusive of ϵ_0. Since $M<N$, it is always necessary to augment the ϵ columns with at least one α β column; hence, complete optimality can never be guaranteed. As will be shown below, the α or β augmenting columns can be determined with the aid of the "ϵ matrix"—the portion of matrix (10) to the right of column ϵ_0. As an example, (11) shows the ϵ matrix for the problem of Fig. 2.

$$\begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \begin{array}{c} \begin{array}{ccc} 1 & 2 & 5 \end{array} \\ \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ -1 & 1 & 0 \\ -1 & 1 & 0 \\ 0 & 1 & -1 \end{bmatrix} \end{array}. \tag{11}$$

Any $M\times M$ nonsingular matrix constructed by deleting $N-M$ rows from the ϵ matrix will be called a "reduced ϵ matrix." If rows $i, j, \cdots$ are the rows deleted from the ϵ matrix to form the reduced matrix, then α_i or β_i, α_j or β_j, $\cdots$ are the columns to be attached to the ϵ columns to form a nonsingular matrix. Consequently, α_i or β_i, α_j or β_j, $\cdots$ are the scales which should be left unspecified, while the remaining α's and β's are the scales which can be independently prescribed. Since the α's and β's play identical roles with respect to the scaling process, it is immaterial whether α_i or the β_i is left unspecified. For simplicity, therefore, it will be assumed that at the outset all the β scales are specified according to the criterion of (2).

$$\begin{array}{c} \\ 1 \\ 2 \\ 3 \end{array} \begin{array}{c} \begin{array}{ccc} 1 & 2 & 5 \end{array} \\ \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 1 & -1 \end{bmatrix} \end{array}. \tag{12}$$

Matrix (12) represents a possible reduction of (11). In this example α_3 and α_4 are to be left unspecified. After specifying ϵ_0, α_1, α_2, α_5 and all the β's, (8) can be used to solve for ϵ_1, ϵ_2 and ϵ_5, and subsequently for α_3 and α_4. After determining ϵ_3 and ϵ_4 through (7), all the γ's and δ's can be found through (5) and (6) respectively.

The generalized outline for the above procedure is:

1) Using the topological constraints of types 1) and 3), eliminate from the equilibrium constraints all γ's and redundant ϵ's.
2) Form the ϵ matrix.
3) Find a reduced ϵ matrix.
4) Specify ϵ_0, all β's, and those α scales which are indicated by the rows of the reduced matrix. ϵ_0 is to be specified according to the desired computation speed, and the α's and β's according to the criteria of (2) and (3), respectively.
5) Using the specified quantities with the ϵ matrix and the topological constraints of type 3), evaluate all ϵ's and unspecified α's.
6) Using the topological constraints of types 1) and 2), evaluate all γ's and δ's.

MANIPULATIONS OF THE ε MATRIX

Every row in the ϵ matrix contains at least two unities (positive or negative); the rest of the elements are zero. Since there is always an independent variable, there will be at least one row containing a single unity. These properties imply that if a nonsingular ϵ matrix exists at all, it can always be found as follows. Select a row with a single unity; then select N-1 rows successively, such that each additional row will contain a unity in exactly one column which is zero in all the previously selected rows. Thus, a reduced ϵ matrix can be found directly with no need for exhaustive search. It can also be seen that, once the reduced matrix is constructed, evaluating the ϵ's does not entail simultaneous solution, but can be done recursively by proceeding from one row to the next, in the order of their selection.

In most problems the choice of rows in the above reduction scheme is not unique, in which case more than one set of specifiable α's will be available. Correspondingly, there may be several sets of solutions for the α scales. In matrix (11), for example, the selected rows may be 1-2-5, 1-3-5, 1-4-5 (where row 1 is the starting row), 2-3-5, 2-4-5 (where row 2 is the starting row).

The facility in which the reduced ϵ matrix and a corresponding set of scales can be produced is quite advantageous, since no solution is guaranteed to be adequate even if it does satisfy the equilibrium and topological constraints. It may happen that one or more of the unspecified α's come out lower than the value given by (2), in which case overflow will occur before the computation terminates. Additional difficulty may be caused by the fact that the range of the scales is limited by the size of the registers, and that the difference $\alpha_i-\delta_i$ $(i=1, 2, \cdots, N)$ has to exceed a certain bound. In practice, these restrictions, which may be called "boundary constraints," are considerably less severe than the constraints previously discussed, since they involve inequalities rather than equalities. If the boundary constraints are violated by the first reduction of the ϵ matrix, a second one has to be carried out, and the process repeated until these constraints are satisfied. If no reduction yields a satisfactory set of scales, the values specified for the specifiable α's have to be raised, and the entire process repeated. Since no simplified procedure has been found for these cases, the search for scales here has to be done exhaustively.

In the above discussion it was assumed that the ϵ matrix can always be reduced. This assumption is not valid for the relatively rare problems in which the integrating packages can be divided into groups coupled only through dy inputs. When this is the case, it is necessary to substitute one or more of the ϵ columns with α columns before a reduced matrix can be formed. Clearly, not more than M columns need to be replaced under any conditions.

Automatic Scaling Routine

The procedures described in the previous sections can be programmed as a scaling routine to be executed by a digital computer. The initial data required by this routine are the topological interconnections, the desired computation speed, the integrand maxima and the constant multipliers for all the integrating packages. The output is a compatible set of $5N+1$ scales.

The specification of the optimal α scales requires the knowledge of the maxima of all the integrands. Quite often this information is available only after the problem is run on the DDA. This difficulty can be resolved by first guessing the maxima and letting the routine compute a set of scales based on these guesses. After the first problem run, an inspection of all the integrands can serve to improve the previous guesses and consequently to yield more satisfactory scales. After several cycles, the scales will achieve their optimal values, and the DDA its most accurate mode of operation for the given problem. This iterative exchange of information between the scaling routine and the DDA is especially convenient when the analyzer at hand operates in conjunction with a general-purpose computer. Usage of a general-purpose computer for both scaling and problem running is also possible; such an operation, however, is seldom advantageous, since general purpose programs for the solution of differential equations are generally slower and more difficult to compile than corresponding DDA programs.

Conclusion

At present, all scaling operations for DDA's are done manually, by trial-and-error methods. This severely limits the usefulness of large-capacity DDA's (containing 100 or more integrating packages) which are available today. The above discussion shows that a compatible and optimal set of DDA scales can be produced systematically. In many practical problems the systematic scaling is direct and does not require an exhaustive search. In more difficult problems, the searching process can be considerably facilitated by the usage of a general purpose digital computer.

Acknowledgment

The author would like to thank the Bendix Computer Division of the Bendix Aviation Corporation for making this project possible, and Prof. Harry D. Huskey of the University of California for his useful suggestions and continuous encouragement.