The CORDIC Computing Technique

JACK VOLDER†

THE "*CO*ordinate *R*otation *DI*gital *C*omputer" computing technique can be used to solve, in one computing operation and with equal speed, the relationships involved in plane coordinate rotation; conversion from rectangular to polar coordinates; multiplication; division; or the conversion between a binary- and a mixed-radix system.

The CORDIC computer can be described as an entire transfer computer with a special serial arithmetic unit, consisting of 3 shift registers, 3 adder-subtractors, and special interconnections. The arithmetic unit performs a sequence of simultaneous conditional additions or subtractions of shifted numbers to each register. This performance is similar to a division operation in a conventional computer.

Only the trigonometric algorithms used in the CORDIC computing technique will be covered in this paper. These algorithms are suitable only for use with a binary code. This fact possibly accounts for their late appearance as a numerical computing technique. Matrix theory, complex-number theory, or trigonometric identities can be used to prove rigorously these algorithms. However, to help give a more intuitive and pictorial understanding of the basic technique, plane trigonometry and analytical geometry are used in this explanation whenever possible.

First, consider two given coordinate components Y_i and X_i in the plane coordinate system shown in Fig. 1.

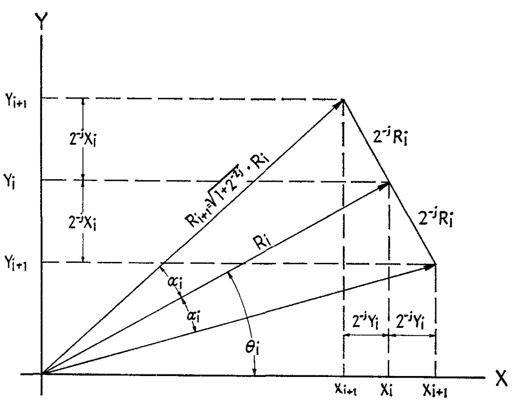

Fig. 1—Geometry of a typical rotation step.

The subscript i, as used in this report, will identify all quantities with a particular step in the computing sequence. The given components, Y_i and X_i, actually describe a vector of magnitude R_i at an angle θ_i from the origin according to the relationship,

$$Y_i = R_i \sin \theta_i \tag{1}$$

$$X_i = R_i \cos \theta_i. \tag{2}$$

With a very simple control of an arithmetic unit operating in a binary code, the sign of a number can be changed and/or the number can be divided by a power of two. Thus, if it is assumed that the numerical values of Y_i and X_i are available, the numerical values of both coordinates of one of the proportional quadrature vectors, R_i', can be easily obtained.

$$Y_i' = 2^{-j}X_i \tag{3}$$

$$X_i' = -2^{-j}Y_i \tag{4}$$

where j is a positive integer or zero.

The vector addition of R_i' to R_i, by the algebraic addition of corresponding components, produces the following relationships:

$$Y_{i+1} = \sqrt{1+2^{-2j}}R_i \sin(\theta_i + \tan^{-1} 2^{-j}) = Y_i + 2^{-j}X_i \tag{5}$$

$$X_{i+1} = \sqrt{1+2^{-2j}}R_i \cos(\theta_i + \tan^{-1} 2^{-j}) = X_i - 2^{-j}Y_i \tag{6}$$

$$R_{i+1} = \sqrt{1+2^{-2j}}R_i. \tag{7}$$

Likewise, the addition of the other proportional quadrature vector at $\theta - 90°$ to the vector R_i produces the following relationships:

$$Y_{i+1} = \sqrt{1+2^{-2j}}R_i \sin(\theta_i - \tan^{-1} 2^{-j}) = Y_i - 2^{-j}X_i \tag{8}$$

$$X_{i+1} = \sqrt{1+2^{-2j}}R_i \cos(\theta_i - \tan^{-1} 2^{-j}) = X_i + 2^{-j}Y_i \tag{9}$$

$$R_{i+1} = \sqrt{1+2^{-2j}}R_i. \tag{10}$$

If the numerical values of the components Y_i and X_i are available, either of the two sets of components Y_{i+1} and X_{i+1} may be obtained in one word-addition time with a special arithmetic unit (as shown in Fig. 2) operating serially in a binary code.

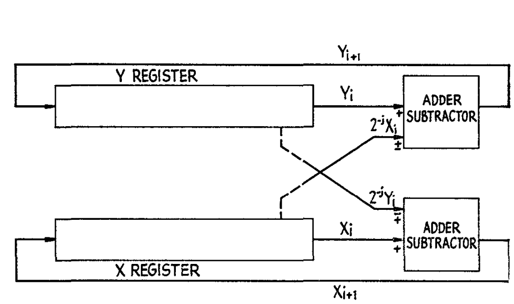

Fig. 2—Arithmetic unit for cross addition.

This particular operation of simultaneously adding (or subtracting) the shifted X value to Y and subtracting (or adding) the shifted Y value to X is termed "cross addition."

† Convair, Fort Worth, Tex.

The effect of either of these two choices can be considered as a *rotation* of the vector R_i through the special angle plus (or minus) α_i where

$$\alpha_i = \tan^{-1} 2^{-j} \tag{11}$$

accompanied by an *increase* in *magnitude* of each component by the factor $(1+2^{-2j})^{\frac{1}{2}}$.

Note that this increase in magnitude is a function of the value of the exponent j and is independent of whichever of the two choices of direction is used. If a particular value of j is specified to correspond to a particular value of i in the general expression, and if it is specified that, for every ith term, one and only one of the two permissible directions of rotation is used to obtain the $i+1$ terms, then the choice may be identified by the binary variable ξ_i where $\xi_i = +1$ for positive rotation, or -1 for negative rotation. This gives a general expression for the $i+1$ terms as

$$Y_{i+1} = \sqrt{1 + 2^{-2j}} R_i \sin (\theta_i + \xi_i \alpha_i) = Y_i + \xi_i 2^{-j} X_i \tag{12}$$

$$X_{i+1} = \sqrt{1 + 2^{-2j}} R_i \cos (\theta_i + \xi_i \alpha_i) = X_i - \xi_i 2^{-j} Y_i \tag{13}$$

$$R_{i+1} = \sqrt{1 + 2^{-2j}} R_i. \tag{14}$$

After the components Y_{i+1} and X_{i+1} are obtained, another similar operation can be undertaken to obtain the $i+2$ terms.

$$Y_{i+2} = \sqrt{1+2^{-2(j+1)}} [\sqrt{1+2^{-2j}} R_i \sin (\theta_i + \xi_i \alpha_i + \xi_{i+1} \alpha_{i+1})]$$
$$= Y_{i+1} + \xi_{i+1} 2^{-(j+1)} X_{i+1} \tag{15}$$

$$X_{i+2} = \sqrt{1+2^{-2(j+1)}} [\sqrt{1+2^{-2j}} R_i \cos (\theta_i + \xi_i \alpha_i + \xi_{i+1} \alpha_{i+1})]$$
$$= X_{i+1} - \xi_{i+1} 2^{-(j+1)} Y_{i+1} \tag{16}$$

$$R_{i+2} = \sqrt{1+2^{-2(j+1)}} \sqrt{1+2^{-2j}} R_i. \tag{17}$$

Likewise, the pseudo-rotation steps can be continued for any finite, pre-established number of steps of pre-established increments but arbitrary values of sign. After these steps have been completed, the increase in magnitude of the vector as a result of these steps will be the constant factor

$$\sqrt{1 + 2^{-2j}} \cdot \sqrt{1 + 2^{-2(j+1)}} \cdot \sqrt{1 + 2^{-2(j+2)}} \cdots \cdot \sqrt{1 + 2^{-2m}}. \tag{18}$$

The effective angular rotation λ of the vector system will be the value of the algebraic summation of the individual rotations.

$$\lambda = \xi_1 \alpha_1 + \xi_2 \alpha_2 + \xi_3 \alpha_3 + \cdots + \xi_n \alpha_n \tag{19}$$

where

$$\alpha_i = \tan^{-1} 2^{-j} \tag{20}$$

and

$$\xi_i = +1 \text{ or } -1. \tag{21}$$

Therefore, although the magnitude of each individual rotation step is fixed, there now appears the possibility that, by an appropriate choice for each ξ, the algebraic summation of all steps can be made to equal any desired angle.

The requirements for making this sequence of steps suitable for use with any angle as the basis of a computing technique are: 1) a value must be determined for each angle α_i so that for any angle θ from $-180°$ to $+180°$ there is at least one set of values for the ξ operators that will satisfy (19), and 2) these chosen values must permit the use of a simple technique for determining the value of each ξ to specify λ.

The following relationships are necessary and sufficient for a sequence of constants to meet these requirements:

$$180° \leq \alpha_1 + \alpha_2 + \alpha_3 + \cdots + \alpha_n + \alpha_n \tag{22}$$

$$\alpha_i \leq \alpha_{i+1} + \alpha_{i+2} + \cdots + \alpha_n + \alpha_n. \tag{23}$$

The following sequence meets the requirements of (22), (23) and (11):

$$\text{First term:} \quad \alpha_1 = 90° \tag{24}$$

$$\text{Second term:} \quad \alpha_2 = \tan^{-1} 2^{-0} = 45° \tag{25}$$

$$\text{Third term:} \quad \alpha_3 = \tan^{-1} 2^{-1} \approx 26.5° \tag{26}$$

$$\text{General term:} \quad \alpha_i = \tan^{-1} 2^{-(i-2)} \quad (i > 1). \tag{27}$$

Any angle can now be represented by the expression

$$\lambda = \xi_1(90°) + \xi_2 \tan^{-1} 2^{-0} + \xi_3 \tan^{-1} 2^{-1} + \cdots$$
$$+ \xi_n \tan^{-1} 2^{-(n-2)}. \tag{28}$$

The combination of values of the operators $\xi_1 \xi_2 \cdots \xi_n$ form a special binary code which is based on a system of *A*rc *T*angent *R*adices and will be identified as the ATR code. The values of α selected for this computing technique will be called ATR (Arc Tangent Radix) constants.

In addition, only one more term is required in this ATR system than that required in a perfect binary-radix system for equivalent angular resolution;

$(n-1)$th term of perfect binary system ($\pm$variable)

$$= 2^{-n} \text{ revolutions} \tag{29}$$

nth term of ATR system (for large n)

$$\approx \frac{2^{-(n-1)}}{\pi} \text{ revolutions} \tag{30}$$

$$\frac{2^{-(n-1)}}{\pi} < 2^{-n}. \tag{31}$$

Note that all terms except the first are terms of the natural sequence $\tan^{-1} 2^{-j}$ ($j = 0, 1, 2$, etc.) and may be instrumented as shown in Fig. 2.

The computation step corresponding to the most significant radix is simply

$$Y_2 = R_1 \sin (\theta_1 + \xi_1 90°) = \xi_1 X_1 \tag{32}$$

$$X_2 = R_1 \cos (\theta_1 + \xi_1 90°) = -\xi_1 Y_1 \tag{33}$$

where

$$R_2 = R_1. \quad (34)$$

This step is unique in that no magnitude change is introduced. It may be instrumented with the same circuitry required for all of the other steps by simply disabling the direct input to the adder-subtractor during this step.

The change in magnitude of the components resulting from the use of all of the terms in the series of (28) is the constant factor

$$\sqrt{1 + 2^{-0}} \cdot \sqrt{1 + 2^{-2}} \cdot \sqrt{1 + 2^{-4}} \cdots \cdot \sqrt{1 + 2^{-2(n-2)}}. \quad (35)$$

The value of this magnitude factor is a function of n which can be a constant for any given computer. By arbitrarily solving for the factor for $n = 24$ and by denoting the value of the magnitude change factor as K,

$$K = 1.646760255. \quad (36)$$

At this point, these individual steps can be fitted into a complete computing technique. Of the two basic algorithms that will be described here, the problem of "vectoring" will be considered first. Vectoring is the term given to the conversion from rectangular-coordinate components to polar coordinates, that is, given the Y and X components of a vector, the vector magnitude R and its angular argument θ are to be computed. In this technique, R and θ are computed simultaneously and in separate register locations.

First consider the problem of computing R. Except for the known magnitude change K, the Pythagorean relationship of the coordinate components is maintained regardless of the value of the summation of rotation angles, λ. Then, if the individual directions of rotations, ξ_i, can be controlled so that, after the end of the computing sequence, the Y component is zero and the X component is positive,

$$R_{n+1} = \sqrt{X_{n+1}^2 + Y_{n+1}^2} = X_{n+1} = KR_1 = K\sqrt{X_1^2 + Y_1^2}. \quad (37)$$

The technique for driving θ to zero is based on a numerical nulling sequence similar to nonrestoring division.

Since the vector R_i is described only in terms of its rectangular-coordinate components, the angle of this vector θ_i from the origin (positive X axis) is not known. However, if θ_i is considered to be expressed in a form so that

$$-180° \leq \theta_i < 180°, \quad (38)$$

then it can be shown that the sign of the Y_i component always corresponds to the sign of the angle θ_i.

Therefore, before each step of the computation, the sign of Y_i may be examined to determine which of the two possible values of ξ_i will drive θ_{i+1} opposite in direction from θ_i and then set the action of the adder-subtractor accordingly to obtain the relationship

$$|\theta_{i+1}| = \big|\,|\theta_i| - \alpha_i\big|. \quad (39)$$

Regardless of the choice for ξ_i, the shift gates and the register gates can be controlled so that the increments of rotation prescribed by the sequence of ATR constants are used in the same order (most significant first) as shown in (24)–(27).

It can be shown that, by adding another term α_n to the summation of all ATR constants, the summation is greater than or equal to 180° for any value of n:

$$180° \leq \alpha_1 + \alpha_2 + \alpha_3 + \cdots + \alpha_n + \alpha_n. \quad (40)$$

By expressing the angle of any vector in the form given by (37), the following relationship is obtained:

$$|\theta_i| \leq \alpha_1 + \alpha_2 + \alpha_3 + \cdots + \alpha_n + \alpha_n. \quad (41)$$

Although it has been previously stated, without proof, it can be readily shown that, for the ith term,

$$\alpha_i \leq \alpha_{i+1} + \alpha_{i+2} + \cdots + \alpha_n + \alpha_n. \quad (42)$$

Therefore, if the same rules given for (39) are applied to determine θ_{i+1},

$$-\alpha_1 \leq |\theta_1| - \alpha_1 \leq \alpha_2 + \alpha_3 + \cdots + \alpha_n + \alpha_n. \quad (43)$$

Then, by applying the inequality of (42) to the left-hand term of the above equation,

$$|\theta_2| \equiv \big|\,|\theta_1| - \alpha_1\big| \leq \alpha_2 + \alpha_3 + \cdots + \alpha_n + \alpha_n. \quad (44)$$

Likewise, this process may be continued through α_n to give

$$|\theta_{n+1}| \leq \alpha_n. \quad (45)$$

As an illustration of the step-by-step value of the vector, as described by the coordinate components at each step during the vectoring operation, consider the example in Fig. 3.

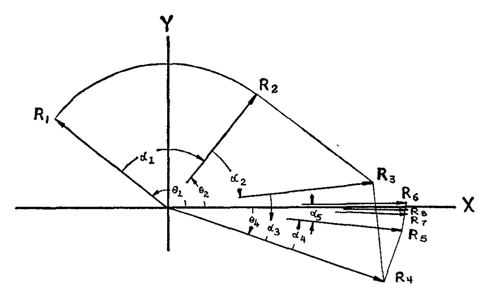

Fig. 3—Step-by-step relationships during nulling.

If, at the end of the computing sequence, the coordinate components specify a θ_{i+1} equal to zero, the total amount of rotation performed was equal in magnitude but opposite in sign to the angle θ_1 as specified by the original coordinate components Y_1 and X_1.

At this point another register (identified as the angle register) and another adder-subtractor may be introduced, and it shall be assumed that the numerical value of each of the preselected ATR constants is stored within the computer and can be made available to the arithmetic unit in the same order as specified in (28). Since each ξ controls the action of the cross addition, each may also control the action of the additional adder-subtractor so that a subtraction or an addition may be made simultaneously in the angle register of the numerical value of the corresponding ATR constant to an accumulating sum to obtain the numerical value of the original angle θ_1. Then, at the end of the computation, the desired numerical value of θ_1 is in this additional register, and the quantity KR_1 is in the X register.

A block diagram of the complete arithmetic unit necessary for computing both R_1 and θ is shown in Fig. 4.

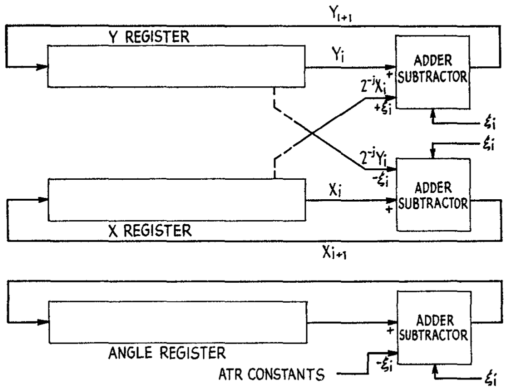

Fig. 4—Block diagram of complete arithmetic unit.

In summarizing the vectoring operations, the initial coordinate components Y_1 and X_1 of a vector are given, and the quantities R_{n+1} and θ_1 are computed where

$$R_{n+1} = KR_1 = K\sqrt{X_1^2 + Y_1^2} \quad (46)$$

and

$$\theta_1 = \tan^{-1} \frac{Y_1}{X_1} \quad (47)$$

and K is the constant magnitude-change factor described in (36) and (37).

A typical computing sequence is shown in Table I. The two's complement rotation is used for all quantities and for simplicity, shifted quantities are simply truncated without round-off.

Next, consider the application of the previous step-by-step relationships for the solution of the problem of coordinate rotation. It will help in applying this technique to consider, as before, the coordinate system as being fixed and the vector as being rotated. Then the solution for coordinate rotation requires the solution of the numerical values of the two coordinate components of a vector that has been rotated by a given

TABLE I

TYPICAL VECTOR COMPUTING SEQUENCE

Y Register	X Register	Angle Register	
$Y_1=$ 0.0101110	1.1000101$=X_1$	0.0000000	
−1.1000101	+0.0101110	+0.1000000	$\tan^{-1}\infty$
0.0111011	0.0101110	0.1000000	
−0.0101110	+0.0111011	+0.0100000	$\tan^{-1}1$
0.0001101	0.1101001	0.1100000	
−0.0110100	+0.0000110	+0.0010010	$\tan^{-1}2^{-1}$
1.1011001	0.1101111	0.1110010	
+0.0011011	−1.1110110	−0.0001001	$\tan^{-1}2^{-2}$
1.1110100	0.1111001	0.1101001	
+0.0001111	−1.1111110	−0.0000101	$\tan^{-1}2^{-3}$
0.0000011	0.1111011	0.1100100	
−0.0000111	+0.0000000	+0.0000010	$\tan^{-1}2^{-4}$
1.1111100	0.1111011	0.1100110	
+0.0000011	−1.1111111	−0.0000001	$\tan^{-1}2^{-5}$
1.1111111	0.1111100$=KR_1$	0.1100101$=\theta$	

angle λ from its original position, as defined by the given initial coordinate components Y_1 and X_1.

The desired angle of rotation will be given in binary coded form. By placing the angle λ in the angle register and sensing the sign of the quantity in this register before each step, the quantity in the angle register may be nulled to zero by sequentially subtracting or adding each of the ATR constants to the remaining quantity.

The explanation and proof of this nulling operation, in which the actual numerical values of the angles are employed, is the same as that for the previous nulling operation for vectoring.

Immediately following the determination of each ATR digit, and concurrently with the operation of the subtraction or addition nulling operation in the angle register, the operation of cross addition of shifted quantities may be performed in the Y and X registers to rotate the vector in a direction determined by each ξ with an angular magnitude as specified by α_i corresponding to the ATR constant being used. Then, at the end of the computing sequence, the numerical values of the desired components Y_{n+1} and X_{n+1} will be in the Y and X registers, respectively.

In summarizing the coordinate-rotation operation, the initial coordinates Y_i and X_i and the desired angle of rotation λ_1 are given where

$$Y_1 = R_1 \sin \theta_1 \quad (48)$$

$$X_1 = R_1 \cos \theta_1. \quad (49)$$

The results available at the end of the computation sequence are

$$Y_{n+1} = R_{n+1} \sin (\theta_1 + \lambda_1) = KR_1 \sin (\theta_1 + \lambda_1) \quad (50)$$

$$X_{n+1} = R_{n+1} \cos (\theta_1 + \lambda_1) = KR_1 \cos (\theta_1 + \lambda_1) \quad (51)$$

where K is the same constant magnitude-change factor, given in (36) and (37), that applies for the vectoring operation.

A typical coordinate rotation computing sequence is shown in Table II. (Note that in this computation, the vector is rotated through the negative X axis.)

Since the change of magnitude will be exactly known beforehand, it may be compensated for exactly either by scaling or by a magnitude-correction multiplication. It may then be said that, except for the practical consideration of limiting the number of digits and the number of steps to some finite value, both algorithms produce an exact solution. In a practical computer, no approximations are necessary except round-off.

In applying this computing technique to practical problems, the complete solution may be programmed by considering the computer as the digital equivalent of an analog resolver.

If the analog-resolver solution flow for the particular problem is known, then the number of operations and the information-flow diagram can be obtained simply by substituting for each resolver, on a time-shared basis, a CORDIC operation.

TABLE II
TYPICAL ROTATION COMPUTING SEQUENCE

Y Register	X Register	Angle Register	
$Y_1=$ 0.0101110	1.1000101 $=X_1$	0.1100101 $=\lambda$	
+1.1000101	−0.0101110	−0.1000000	$\tan^{-1}\infty$
1.1000101	1.1010010	0.0100101	
+1.1010010	−1.1000101	−0.0100000	$\tan^{-1} 1$
1.0010111	0.0001101	0.0000101	
+0.0000110	−1.1001011	−0.0010010	$\tan^{-1} 2^{-1}$
1.0011101	0.1000010	1.1110011	
−0.0010000	+1.1100111	+0.0001001	$\tan^{-1} 2^{-2}$
1.0001101	0.0101001	1.1111100	
−0.0000101	+1.1110001	+0.0000101	$\tan^{-1} 2^{-3}$
1.0001000	0.0011010	0.0000001	
+0.0000001	−1.1111000	−0.0000010	$\tan^{-1} 2^{-4}$
1.0001001	0.0100010	1.1111111	
−0.0000001	+1.1111100	+0.0000001	$\tan^{-1} 2^{-5}$
1.0001000	0.0011110	0.0000000	

Monte Carlo Calculations in Statistical Mechanics

W. W. WOOD† AND J. D. JACOBSON†

I. Statistical Mechanical Introduction

According to classical statistical mechanics, the thermodynamic properties of a system of N molecules at temperature T and volume V are obtainable from the Gibbs configurational phase integral,

$$Z_N(T, V) = \int_V \cdots N \cdots \int_V e^{-U/kT} d\mathbf{r}_1 \cdots d\mathbf{r}_N, \qquad (1)$$

where k is Boltzmann's constant; U is the potential energy of the system of N molecules, and is a function of the position vectors $\mathbf{r}_\alpha$, $\alpha = 1, 2, \cdots, N$, of the molecules. For suitably simple molecules one usually assumes U to be expressible as a sum of spherically symmetric pair interactions $u(r)$:

$$U(\mathbf{r}_1, \cdots \mathbf{r}_N) = \frac{1}{2} \sum_{\alpha=1}^{N} \sum_{\beta=1}^{N}{}' u(r_{\alpha\beta}), \qquad (2)$$

where $r_{\alpha\beta} = \|\mathbf{r}_\beta - \mathbf{r}_\alpha\|$, and the prime indicates omission of terms for which $\alpha = \beta$.

Most of the thermodynamic functions are expressible in terms of "ensemble averages" of some related function of the configurational coordinates. In these averages the factor $e^{-U/kT}$ appears as a weighting factor, so that Z_N given by (1) is the associated normalizing factor. For example, the pressure p is given by the average of the "virial" V_R of the total intermolecular force:

$$pV/NkT = 1 - (1/3NkT)\langle V_R \rangle, \qquad (3)$$

where

$$\langle V_R \rangle = (1/Z_N) \int_V \cdots N \cdots \int_V \left(1/2 \sum_\alpha \sum_\beta{}' r_{\alpha\beta} du(r_{\alpha\beta})/dr_{\alpha\beta} \right) \times e^{-U/kT} d\mathbf{r}_1 \cdots d\mathbf{r}_N \qquad (4)$$

The "radial distribution function" $g(r)$ is of considerable importance in the study of fluids. Let us first define $n(r)$, the "cumulative radial distribution function," giving the average number of molecules lying within the distance r from any representative molecule:

$$n(r) = (1/N) \left\langle \sum_\alpha \sum_\beta{}' A(r_{\alpha\beta}, r) \right\rangle, \qquad (5)$$

where $A(x, r)$ is the step function

$$\left. \begin{aligned} A(x, r) &= 1 \qquad 0 \leqq x < r \\ &= 0 \qquad r \leqq x \end{aligned} \right\}. \qquad (6)$$

Then $g(r)$, which is the number density of molecules at the distance r from any representative molecule, relative to the over-all macroscopic density N/V, is given by

† Los Alamos Scientific Lab., Los Alamos, N. M.