DECUS

PROGRAM LIBRARY

DECUS NO. 8-559

TITLE CUBIC - A Digital Program for On-Line Differentiation of Sampled Analog Signals

AUTHOR John H. J. Allum

COMPANY Man-Vehicle Laboratory
Massachusetts Institute of Technology
Cambridge, Massachusetts

DATE January 4, 1973

SOURCE LANGUAGE PAL

CUBIC.

A digital program for on-line differentiation of sampled analog signals.

by

John H.J. Allum.

Man-Vehicle Laboratory
Massachusetts Institute of Technology
Cambridge, Massachusetts, 02139.

Financial support.

This work was supported in part by NASA Grant NGR-22-009-025

Abstract.

A digital computer program CUBIC has been developed for on-line differentiation of analog voltage signals. The program accepts voltage records of a time function and yields its derivative after one program cycle time of 4.2 msecs. The velocity is obtained by employing a least mean squares cubic fit technique.

The routine is intended for experimental work either as a data reduction tool or as a control signal for a closed loop experiment. The program can be implemented on a PDP-8 digital computer with one digital to analog converter channel and one analog to digital converter channel.

The derivative (hereafter called the velocity) of an analog signal or a continuous time function (hereafter called the position) is of interest in many classes of research. This information can be obtained by a variety of techniques. The approach used in this program is to fit a cubic curve through the last 33 samples of data using a least mean squares fit technique and taking the slope of this curve at the latest data point. The program developed produces the least noise of all differentiation techniques considered (a six point difference method, pseudo differentiation with a transfer function $\alpha s/(s+\alpha)$, and a parabolic least mean squares technique) and yet is applicable over a wide range of frequencies: d.c. to 8 Hz. The cubic fit algorithm, together with examples of its operation, is described in reference 1.

Operation of the program.

The program is designed for operation in a hybrid installation. The hybrid equipment used consisted of a Digital Equipment Corporation PDP 8 digital computer (4096, 12 bit words, 1.5 μsec. cycle time with Extended Arithmetic Element Type 182), connected to a GPS-290T analog computer with ±10 volts reference voltage. An analog clock generated the sampling signal for the A/D and D/A conversions at a rate of 160 per sec (or every 6.25 msecs.)

Storage.

The digital program is compiled as an eight page subroutine and is listed in Appendix A. The program occupies locations (20,34), (47,111), 126, 127, (5233,5334) and (5600,7567). The auto-index 17 is used and the 33 data points are stored in locations (6000,6040). Double and single precision multiplication routines for signed numbers are required by the routine (ref.2).

Precalculated constants.

The digital program has been compiled assuming 33 data points. Assume N is the number of data points. The constants N, (N-1)*(N-2), and $(N-1)^3$ are placed in locations 6377, 6376 and 6452 respectively. The constant $(N-1)^2$ is formed in instructions 6311 and 6312. Choices other than 33 data points are permissable. 2^r+1, where r is an integer, is an advantageous choice given the binary basis of digital computers.

The values of recursion formulas used in the program, for a unit step in position of duration equivalent to the N data points, are stored in locations (7534, 7540). These values are;

N, $\quad \sum_{i=0}^{N-1} (-i)^2$ lower part,

$\sum_{i=0}^{N-1} (-i)$, $\quad \sum_{i=0}^{N-1} -i(i-1)$ lower part,

$\sum_{i=0}^{N-1} (-i)^3$ respectively.

The upper part of $\sum_{i=0}^{N-1} (-i)^2$ is formed at location 7456, and the upper part of $\sum_{i=0}^{N-1} -(i-1)$ at location 7476. $\sum_{i=0}^{N-1} (2i-1)$ is computed by instructions (7520, 7524).

The velocity estimation equation prior to digital filtering may be written as

$$a_1^n = XVEL = [\ 10\alpha_1 A^n + 10\alpha_2 B^n/\Delta t + 10\alpha_3 C^n/\Delta t^2 + 10\alpha_4 E^n/\Delta t^3\ ] * [\ 1/10\Delta t\ ] \quad (1).$$

Δt is the sampling interval. The constants $10\alpha_1$, $10\alpha_2$, $10\alpha_3(2^4)$, and $10\alpha_4(2^7)$ are precalculated in double precision arithmetic and located at (6750, 6757). The constants α_i (i=1,4) and the recursion formulas A^n, B^n, etc. are defined in reference 1. Both parts of these double precision fractions must be expressed as a signed fraction. For example, the fraction

0.12515777_8

is expressed as

0.1252_8 and -0.00002001_8

and stored as

0.1252 and -0.2001 .

The data points may be stored in any N consecutive core locations. Locations 7565 and 7566 must contain the last and first data storage locations respectively. The order of storage is inverted with respect to sampled time.

Calling sequence.

The program is called as a subroutine. A typical calling sequence including the A/D and D/A conversions is listed in Appendix A. X2 is the filtered estimate of the velocity.

Flow of operations.

Figure 1 is a flow diagram describing the sequence of operations.

-- the AC must be clear at entry, the link status is irrelevant.

-- to restart estimation, START (location 52) must contain a negative value.

-- the jump is made to the subroutine with the latest data value at MTHETA (location 55).

-- the recursion formulas A^n, B^n, etc. are updated.

-- $a_1^n \Delta t$ is calculated in triple precision arithmetic.

-- the unfiltered velocity estimate, a_1^n , is calculated by shift operations, simultaneously a scaling by 2 can occur.

-- the jump is made to the subroutine LIMIT.

-- the jump is made twice to the subroutine FILTER
-- the jump is made to the subroutine LLAGS
-- the filtered velocity estimate is stored at X2 (location 126)
-- the jump is made back to the main program.
-- the AC is clear at exit, the link status is undetermined.

<u>The calculation of a_1^n from $a_1^n\Delta t$.</u>

In the flow diagram (figure 1) $B^n/\Delta t$ is written as B(n), similarly for C(n), etc. At location 6732 the first factor within square parantheses in equation 1 above is contained in four locations designated FVH, FVM, FVL, and FVLL. The binary point is between bit 1 and bit 2 of the register FVM. (The correspondence between the analog voltage reference levels and the digital signed fraction system is 0.3777_8 = 10 volts on the above cited hybrid installation.)

$10\Delta t a_1^n$ is independent of Δt, thus the program operator may choose any scaling factor or sampling rate to obtain the velocity estimate in operations subsequent to that of location 7004. Multiplication by $1/10\Delta t$ is affected by choice of the variable SCALE (location 7045). The subroutine SHIFT shifts the contents of FVH, FVM, FVL, and FVLL left or right to obtain the required multiplication. In the present application one shift left is required to place the decimal point between bit 0 and 1 of register FVM and as Δt equals 6.25 msecs. 4 further shifts are required for multiplication by $1/10\Delta t$. Thus the total content of SCALE is +5. It is recommended that sampling rates which are a binary multiple or divisor of 160 per sec. be used. Storage of the velocity estimate on return from the subroutine SHIFT is at location 7324 (ULOAD). If the estimate exceeds ±10 volts the program halts at location 7016 or 7020.

<u>The subroutine FILTER.</u>

Differentiation routines are very sensitive to high frequency noise. To reduce this noise two equal cascaded simple lag filters are used. The input to the filter cascade is ULOAD and the output is XVEL (location 56). FILTER is one of the lag filters, $(\alpha/(s+\alpha))$, with a cut-off frequency determined by the subroutine LIMIT. The digital filter is based on the state space representation of a simple lag:

$$u(t_i) = \Phi(t_i,t_{i-1})u(t_{i-1}) + \alpha \int_{t_{i-1}}^{t_i} \Phi(t_i,\tau)c(\tau)d\tau,$$

where $u(t_i)$ is the output of the filter and $c(\tau)$ is the input. As a difference equation

$$u(i) = \Phi(\Delta t)u(i-1) + \alpha c(i-1) \int_{t_{i-1}}^{t_i} \Phi(t_i,\tau)d\tau$$

The terms $\Phi(\Delta t)$ and $\alpha \int_{t_{i-1}}^{t_i} \Phi(t_i,\tau)d\tau$ are stored at PHIX and PHIV respectively (7376 and 7375). The program is compiled with α preset at 20.4 Hz.

The subroutine LIMIT

A flow chart for the subroutine LIMIT is shown in figure 2. The function of the subroutine is to determine an appropriate lag constant α for the subroutine FILTER. This is accomplished by determining the maximum velocity (i.e. of XVEL) and maximum input position (i.e. of MTHETA) to obtain an estimate of the fundamental input frequency. Based on this estimate the lag constant is set as follows:

8 when $\omega<1$,
8ω when $64>\omega>1$,
512 when $\omega>64$

where ω is the frequency in radians per second.

If the velocity estimate was a single sinusoidal time function with no attenuation with respect to the true velocity of the input position sinusoidal time function, the α chosen by the subroutine LIMIT would introduce a phase lag of 11.4 degrees into the velocity estimate. In practice the velocity estimate is noisy; the phase lag introduced is about 10 degrees in the range $\omega<64$. A fixed lag constant may be used by setting the variable HOLD (location 6161) to $\alpha/8$ and noting that the binary point lies after bit 11.

Bode plots.

Bode magnitude and phase plots for CUBIC are shown in figures 3 and 4. The -3 db. point, with respect to a pure differentiator, is at 9 Hz., whereas the 45 degrees phase point is at 4.5 Hz. X2 is the filtered estimate of the velocity, being the output of the subroutine LLAGS. LLAGS is a lag-lead digital filter which corrects for high frequency amplification in the range 2.5 to 9 Hz. This amplification is a direct result of the least mean squares cubic fit procedure.

References

Ref. 1. Allum, J.H.J. (1972). A least mean squares cubic fit algorithm for on-line differentiation of sampled analog signals. To be published.

Ref. 2. Double and single precision multiplication routines have been developed in the Man-Vehicle Lab by Dr. Noël A.J. Van Houtte.
"Display Instrumentation for V/STOL Aircraft in Landing", Vol 3, Sc.D. Thesis, MIT, June 1970.

Acknowledgements

The author wishes to acknowledge the helpful advice given by his colleagues Charles Oman, Charles Ormsby, John Tole and Syozo Yasui.

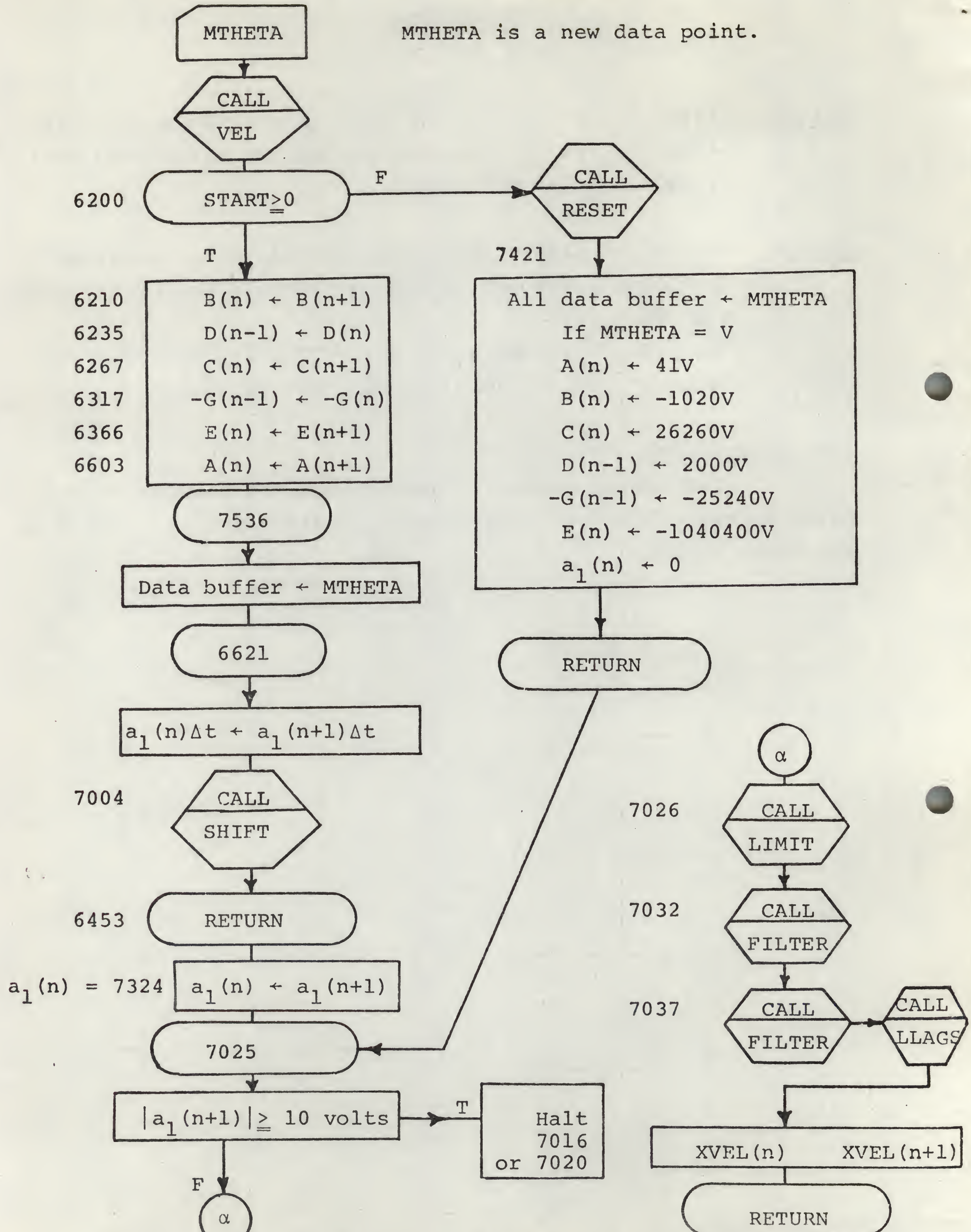

Figure 1. Flow Chart for CUBIC.

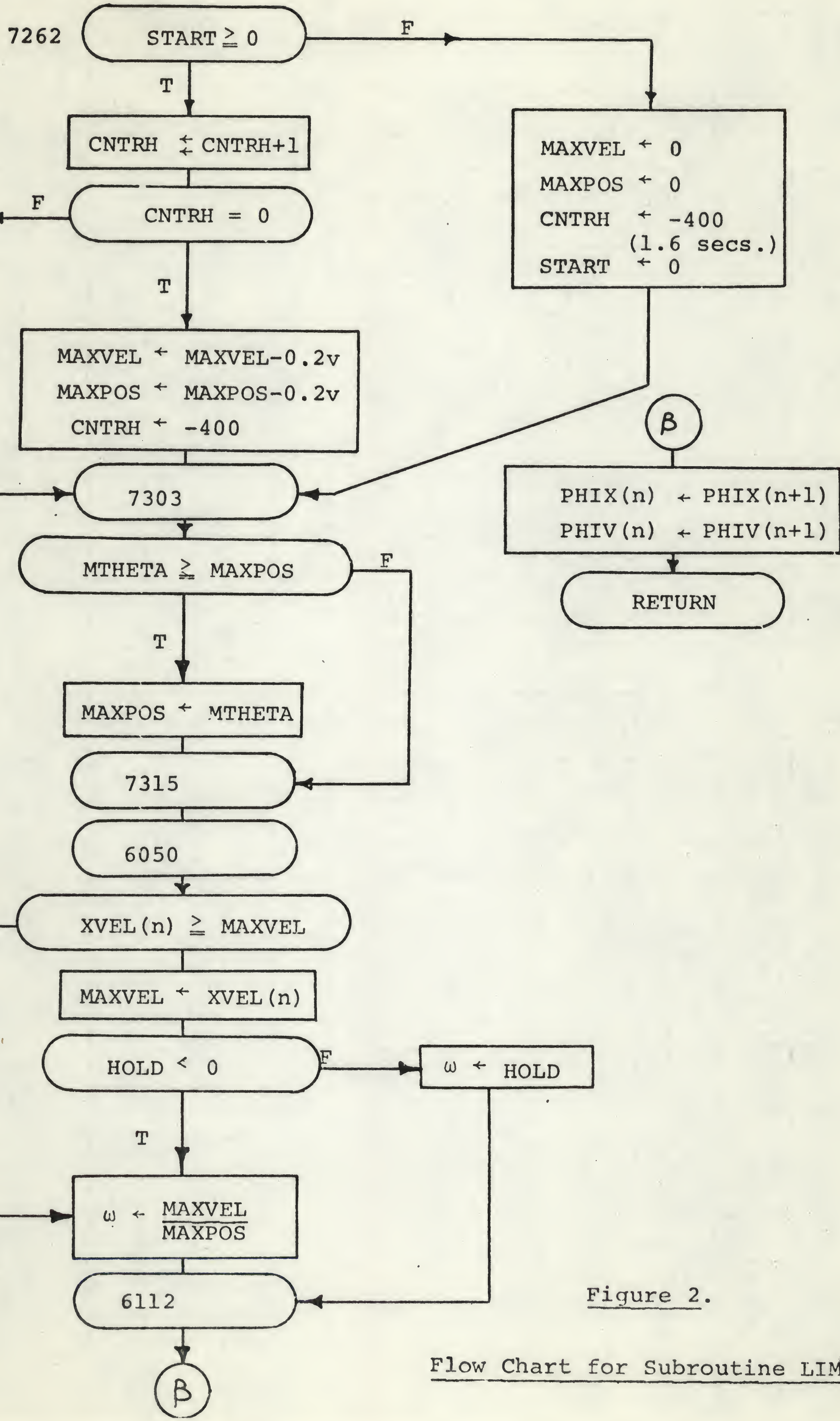

Figure 2.

Flow Chart for Subroutine LIMIT.

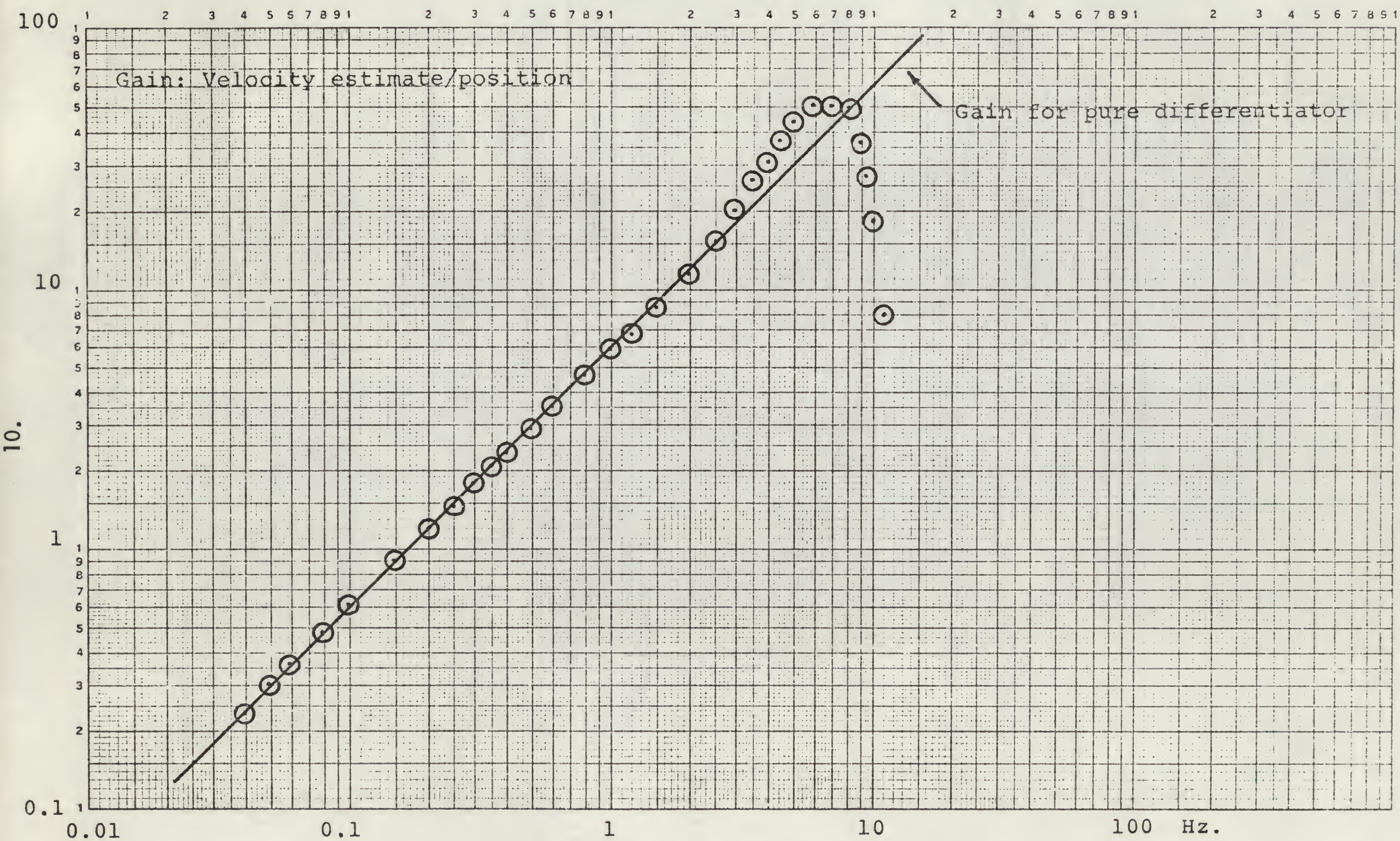

Fig. 2. Bode magnitude plot for CUBIC.

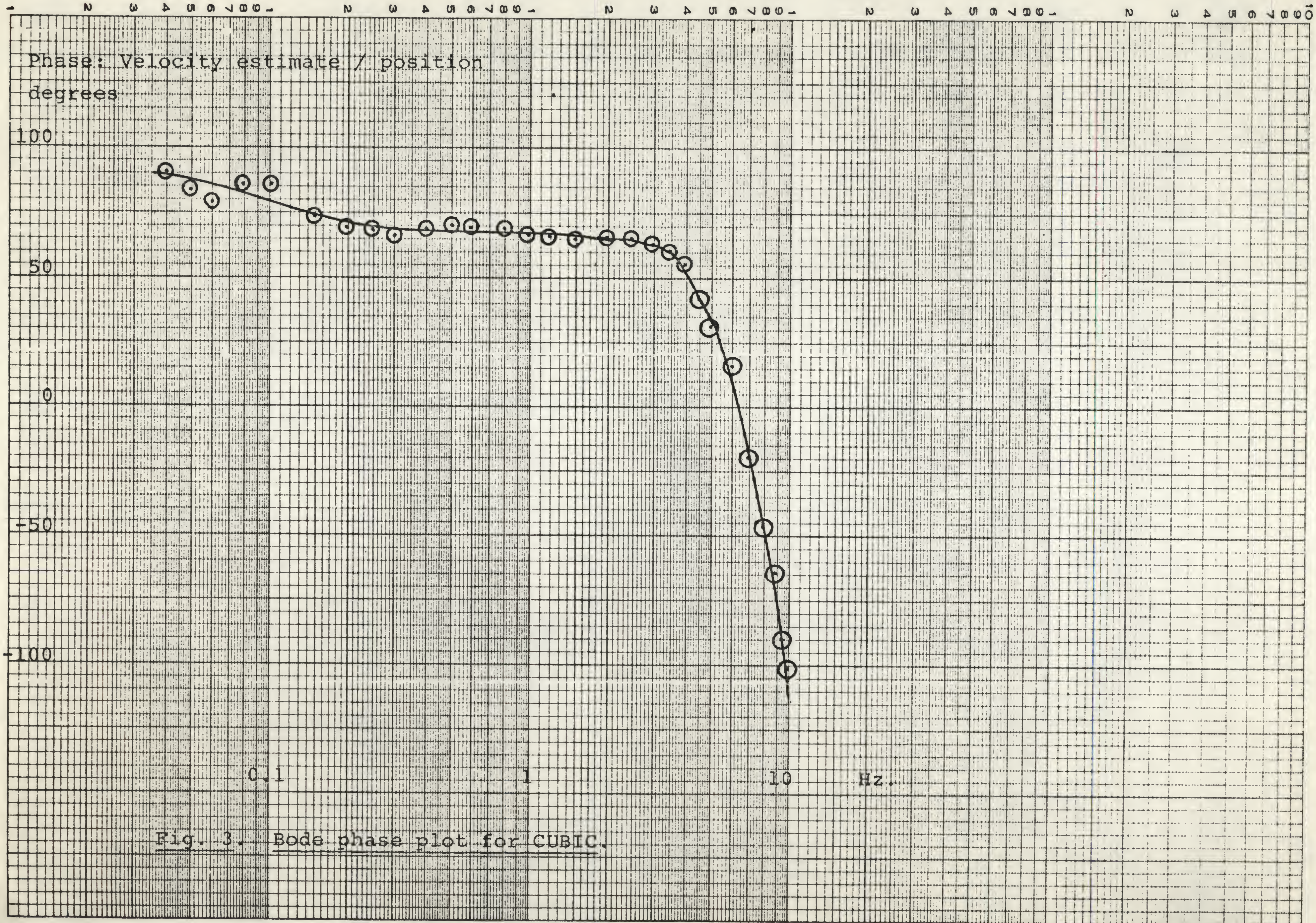

Fig. 3. Bode phase plot for CUBIC.