1. NUMERICAL AND ADAPTIVE CONTROL FOR DIE SINKING

J. Tlusty, Y. Koren, P. McNeil
Department of Mechanical Engineering
McMaster University, Hamilton, Ontario, Canada

Abstract

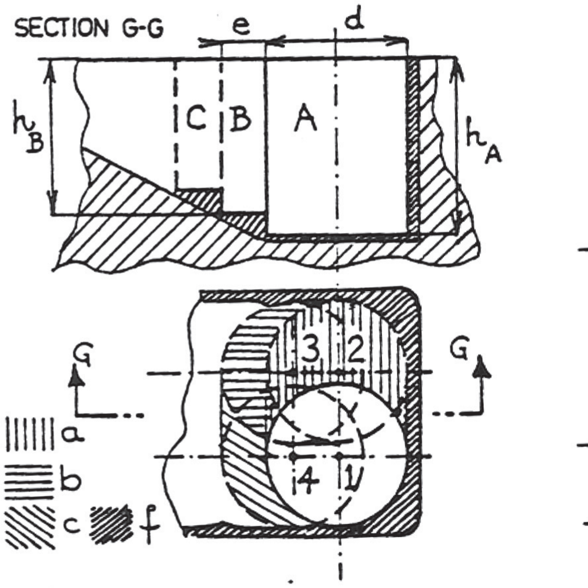
An experimental die sinking machine has been built as a retrofit of a vertical milling machine controlled from an HP 2100A minicomputer. The project of NC and AC for die sinking consists of the development of a cutting force and torque transducer of an investigation into strength limits of end milling cutters and of development of the NC and AC software. The solutions are discussed and programs described.

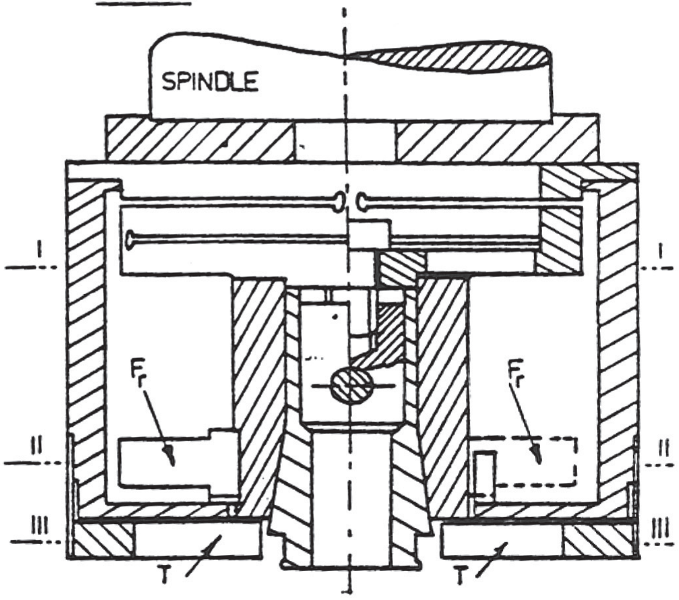
Of all the possible applications of Adaptive Control in machining the one concerning the die sinking is perhaps the most obvious and straightforward. It is a natural extension of the long established practice in copy milling machines where feed-rate used to be controlled from a sensor of the load on the cutter.

The reasons for this application of Adaptive Control are, on one side, the practically unpredictable and usually large variation of the load on the cutter, and on the other side, the vulnerability of the usually long and slender end milling cutters used in die work.

Let us consider an elementary example in Fig. 1 of a rectangular cavity with a sloping bottom. Its rough machining starts with drilling a flat bottommed hole in position 1. A cutter with a comparatively large diameter is used first starting from this hole, moving to Pt 2, Pt 3, Pt 4, etc. The cut Aa represents great load by milling width d and depth h_A. The next move b from Pt 2 to Pt 3 goes in lesser depth h_B and still full diameter width. The section C from Pt 3 to Pt 4 represents milling width e which is less than d/2. Cuts B, C, D, E follow with width e and decreasing depth of cut. After this operation stock f is left to be removed by a smaller diameter cutter in the corners and by a ball ended cutter on the steps of the sloped bottom. It is obvious that loads on these cutters will vary again considerably. In actual practical cases of dies the situation is usually much more complicated than in this simple example which however illustrates well the point concerned.

A project is carried out in the McMaster Metal Cutting laboratory in developing an AC system in addition to a minicomputer NC control of an experimental vertical milling machine. The project consists of three parts: the development of the AC transducer, the determination of strength limits of end milling cutters and the develcoment of the AC and NC software.





The end milling cutter may break in bending or in twist or its tooth may break away. Thus, we have to continuously check the radial cutting force F_r and the cutting torque T on the cutter and vary the feed-rate so as to keep both these variables below the permissible limits. For maximum productivity we should use the maximum feed-rate which satisfies this condition. The transducer for measuring F_r and T must be sensitive and rigid at the same time. Several alternative designs of the transducer are being considered. One of them is indicated in Fig. 2. The transducer is built as a tool holding body to the spindle flange. The collet is clamped in a sleeve which is connected to the outer body in planes I and II. The connection in plane I is radially rigid and it is flexible both in torsion around the cutter axis and in angular tilt out of this axis. The latter flexibility is achieved by means of two slots visible in the drawing. They act as a ball joint in the intersection of plane I with the axis of rotation. In plane 2 elements are located which are sensing the two components of the radial force while still permitting torsional flexibility. The torque T is measured in plane 3 by means of driving arms attached to a radially floating ring. This design was optimized for radial and torsional stiffness with required sensitivity constraints.

The problem of the strength limits of each milling cutters is complicated in its part of tooth breaking which requires three-dimensional stress analysis carried out using finite element computation. Another difficult aspect is that of permissible stress because mostly we have to deal with brittle fracture. This part of the project is still in the initial stage only.

General Description

The block diagram of the system is shown in Fig. 3. It includes five major components: a milling machine, a mini-computer (H.P. 2100A), a data conversion system, a Time Base Generator (TBG) and a Controller which contains 3 control circuits for the 3 axes-of-motion of the milling machine.

The milling machine is equipped with DC servomotors as feed drives, resolvers as positional feedback elements and DC tachogenerators as velocity feedback devices. A special sensor for on-line measurements of the spindle torque, T, and of the cutting force, F, is attached to the spindle. In order to determine the force, two perpendicular cutting forces, F_u and F_v, are sensed. The sensor outputs are fed through an Analog-to-Digital Processor (ADP) to the computer. The computer handles two programs: NC and adaptive control (AC) programs. The interrupt system of the computer takes care of the simultaneous running of both programs. The AC program accepts the sensor outputs and uses them to calculate a feed-rate correction which is supplied to the NC program. The corrections are provided continuously throughout the cutting process in order to maintain maximum feed-rate compatible with the limitations imposed by the strength of the cutter. Normally the computer carries out the AC program, and whenever an interrupt occurs, the JSB instruction (JSB=Jump to Subroutine) causes the computer to transfer its control to the NC program. This instruction also automatically saves the return-address for a later return to the AC program, thus when the NC program is terminated the computer continues to perform the AC program from the point at which it was interrupted.

The computer is equipped with three types of lines: Interrupt input line, Digital output line, Digital input line.

The digital output lines are used for transferring data from the computer through the Controller to the machine drives. For each machine axis two lines are required: one for the sign and the other for command pulses. Each pulse will cause a motion of 0.0001 inch. This unit is the system resolution and will be denoted henceforth as the Basic Length Unit (BLU).

The data for the AC program has to be updated by a continuous information about the cutting process. The measured variables T, F_u and F_v are converted to a binary form by Analog-to-Digital Converters (ADC) and transferred to the computer via the digital input lines.

The three ADC's are included in the ADP, which contains also a Digital scanner. The ADC outputs are selected successively by the Digital Scanner. The data conversion begins simultaneously in the three ADC's, on the trailing edge of a conversion command pulse (CP). The latter is generated by the Data Reading Routine (DRR), which is a part of the AC program. At the completion of a conversion (conversion-time is 20 μ sec) the ADP produces a pulse, which is sent to set the flag of

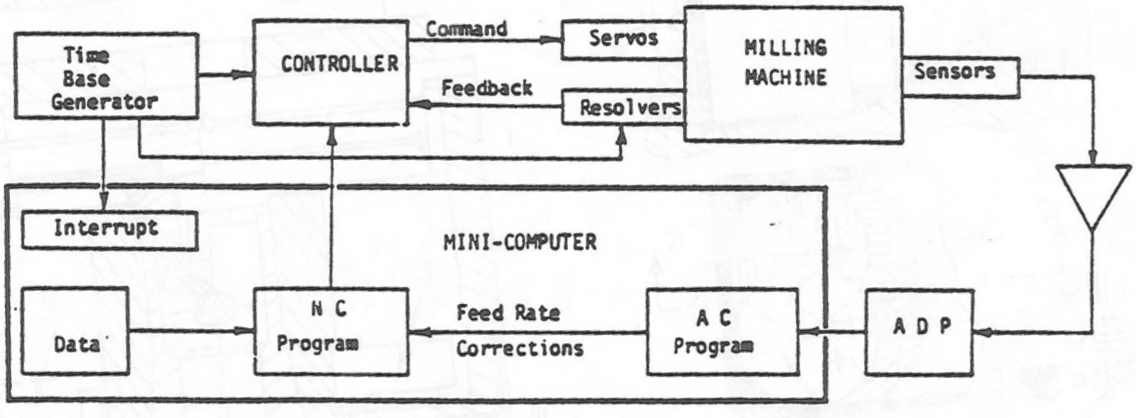


Fig. 3. Block diagram o

DRR 1/0 channel. The set of the flag informs the DRR that the input data is available. The Digital Scanner picks up only one of the ADC outputs at a time and sends it to the computer.

The TBG controls the timing of the systems. It provides the following functions:

1. Supplying clock pulses of 2.5 MHz frequency to the control loops.
2. Providing interrupt pulses to the computer. Each interrupt pulse starts the NC program running. The frequency of the interrupt pulses establishes the maximum feed-rate which can be achieved. This is calculated by using the following formula:
   $$FRM = IPF \times BLU \times 60 \qquad (1)$$
   where FRM is maximum feed-rate (in contouring in ipm.
   IPF - of interrupt pulses frequencies in pps.
   The IPF is limited by the number of instructions in the NC program, and by the instruction execution time. In our system IPF=5000 pps and therefore FRM = 30 ipm.
3. Producing two sine-waves, 90 degrees phase-shifted, which are fed as reference signals to the stators of the resolvers. The frequency of these signals is 2.5 KHz.

In order to synchronize the system, a 2.5 MHz pulse-generator is used as the source for all these signals.

The Controller contains a control circuit for each axis-of-motion. The circuit transforms the train of pulses from the computer command line into a phase-modulated command signal and compares it with the phase-modulated feedback signal. The result is the velocity command, which is fed via an amplification unit to the feed motor.

The positional control is performed by the position loop and software counters, which are contained in the NC program. The counters are loaded with the required incremental distance at the beginning of a segment. Each axis-of-motion is provided with a counter. Each time a command pulse is sent out by the computer, the contents of the appropriate counter are reduced by one unit. The phase-comparators in the control circuits are never in saturation, thus accomplishing the positional control.

The NC Program

The general structure of the NC program is shown in Fig. 4. In this program one can distinguish between the "frame" which enables the cooperation between the NC and the AC programs, and the major part which contains five subroutines: PTP, Feed, Interpolator, Output and Position. The NC program uses an additional routine denoted as the Initiator Routine. The main function of the Initiator Routine is the loading of a new data block to the memory locations which the NC program is using. The "frame" includes the following:

1. When the interrupt occurs, the contents of the arithmetic-unit registers are stored in the computer memory. This stored data is required for the continuation of the AC program.
2. After executing the major part of the program, the stored values are returned and then,
3. The flag of the interrupt-pulses channel is cleared and
4. Control is returned to the point of interruption in the AC program.

The principles of the five sub-routines in the major part of the Continuator Routine are as follows:

PTP In a point-to-point (PTP) operation the axes move in the highest feedrate. An interrupt pulse activates the PTP routine, and then henceforth it starts to run in a closed loop supplying command pulses, the frequency of which depends on the cycle time of this loop. In our case the loop contains 19 instructions. Since the execution time of each instruction is 2μ sec, the frequency of the command pulses is: $60 \times 10^6/38 = 158 \times 10^4$ pulses/min, which means a rapid traverse of 158 ipm per axis. A subtraction of the appropriate counter by one unit is carried out for each command pulse. An automatic deceleration is accomplished when a counter is almost zero. The PTP deceleration is carried out in three steps, and it is a part of the PTP routine. As long as the deceleration is not actuated the PTP routine continues to supply the high frequency pulses. Once this deceleration is actuated, the program returns to the interrupted control mode. When both counters are reaching zero the program jumps to the Initiator Routine.

FEED The Feed routine generates interpolation commands in a rate dependent on the feed-word (f) in the data block. The maximum rate of interpolation commands is equal to the frequency of the interrupt pulses, i.e., 5000 per second. This is compatible with a feedrate of 30 ipm. If the data block contains a deceleration command, the feedrate decreases exponentially to 20 percent of the programmed feedrate in that block.

INTERPOLATOR Three types of interpolation are available: linear and two circular interpolations. The principle of each interpolator is a simulation of two DDA integrators. The simulated registers of the DDA's are loaded by the Initiator Routine. For every interpolation command which is produced by the Feed routine, a single cycle of the DDA is simulated.

OUTPUT If as the result of a DDA cycle an overflow pulse is generated either in one or two axes, this routine sends a command pulse to the Controller.

POSITION For every command pulse the position counter of the appropriate axis is decremented by one unit. A zero position check of both counters is performed. When both counters are zero, (which means that the machining of the current segment has terminated), the program jumps to the Initiator Routine, in order to load a new block and process its data.

The Interpolator

The interpolator, which is a part of the NC program, simulates a hardware interpolator and a feed-rate control based on DDA integrators. The main clock which controls the hardware interpolator is replaced by the source of the interrupt pulses.

Fig. 5 shows the flow-charts of the feed-rate calculator (a), the linear interpolator (b) and the circular one (c). Simpler DDA's, without incrementation of the y registers, are used in the simulation of the linear interpolator and the feed calculator, while full software DDA's are used for the circular interpolator. The feed-rate number f is fed to y_o. For linear milling with paths x and y along the X and Y-axes, the registers y_1 and y_2 are fed by the numbers x and y respectively. In circular operation the initial values of $i=R \cos \omega t$ and $j=R \sin \omega t$ are fed to the registers y_1 and y_2, respectively.

Denoting the frequency of the interrupt

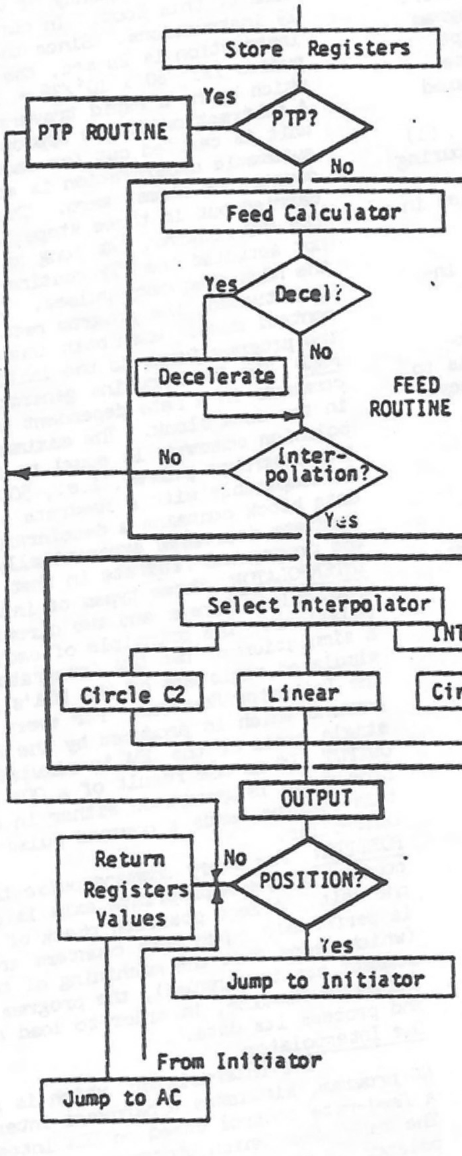


Fig. 4. The NC program.

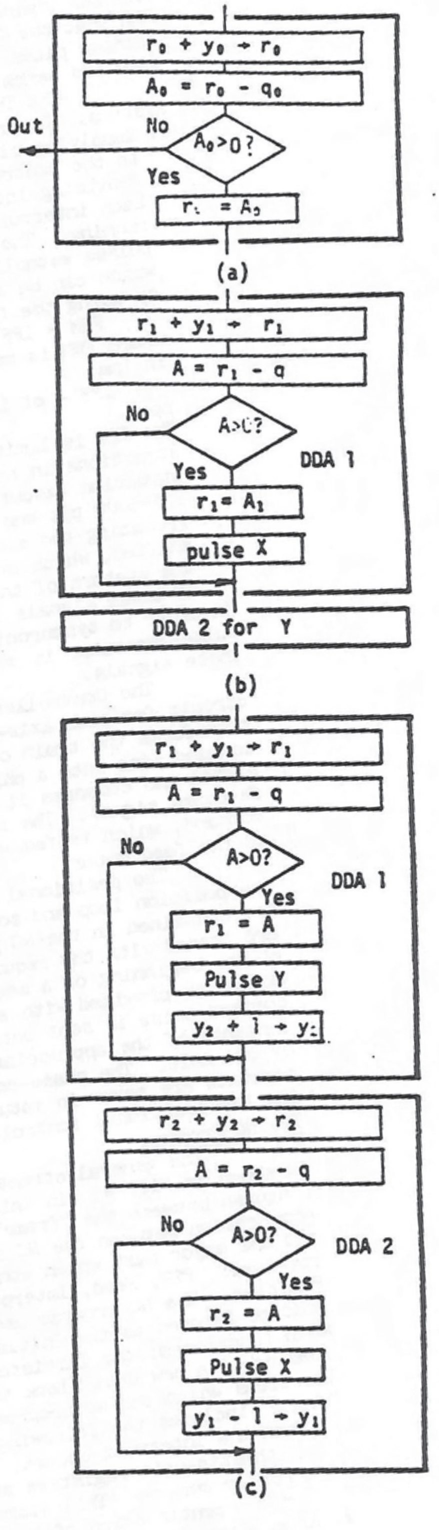


Fig. 5.
a) Feed-rate calculator
b) Linear interpolator
c) Circular interpolator

pulses by W_o, the output frequency of the feed-calculator is $W_o f/q_o$. In linear motion the output frequencies V_x and V_y of the command pulses to the X and Y-axes respectively are:

$$V_x = (W_o f/q_o)\ x/q \quad V_y = (W_o f/q_o)\ y/q \tag{2}$$

In circular motions x and y in Eq. (2) are replaced by i and j, respectively. Obviously the feed-rate along the cutting path is for linear motion

$$V = (W_o f/q_o)\ L/q \tag{3}$$

where $L = \sqrt{x^2 + y^2}$. In circular interpolation the arc radius R replaces the distance L in Eq. (3), where $R = \sqrt{i^2 + j^2}$.

Notice that q and q_o represent in a hardware DDA the maximum (fixed) content of the registers of the interpolator and the Feed Calculator respectively. Since it might happen that the numbers in the y_1 registers are small compared with the value of q the hardware interpolator must use a high frequency clock W_o (in the range of MHz) in order to obtain the required feed-rate. However, it is impossible to use such a high frequency for the interrupt pulses and that leads to other definitions of q. The software interpolator uses a variable q rather than a fixed one. The value of q is calculated at the beginning of each segment. For linear motion: q = L, for circular motion: q = R.
By using these definitions Eq. (3) is reduced to

$$V = W_o f/q_o \tag{4}$$

Since W_o is a constant, the term f/q_o represents the ratio between the required feed-rate and the maximum allowable one. But as this ratio is equal to V/W_o, the frequency W_o is the maximum allowable feed-rate measured in pps. For maximum feed-rate of 30 ipm and a BLU of 0.0001 inch W_o it is 5 KHz. Such a frequency allows time intervals of 200 μ sec for the execution of the NC program. In fact the maximum execution time of the latter is 170 μ sec, which always permits enough time for the execution of the AC program. Another advantage of this interpolator is in calculating the feed-rate number f. By choosing the value of q_o as 3000, any feed-rate until 30 ipm (with a resolution of 0.01 ipm) is programmed directly in ipm by using the formula: f = 100 x FR where FR is the required feed-rate.

AC Program

The AC program consists of three routines: DRR, Error calculator routine (ECR) and Feed-rate calculator (FRC) as is shown in Fig. 6. The DRR receives the input from the ADP and stores it in the appropriate memory locations. With the completion of this stage an output pulse CP is sent and simultaneously the flag in the I/O channel is cleared. Since the conversion time is 20 μsec, a new data will be available, approximately 11 computer instructions after the output pulse. At that time the ADP will send a pulse to set the flag of the I/O channel and a new cycle of the DRR will start.

Approximately every 10 msec the program is switched from the DRR to the ECR. The time measurement is done indirectly by counting the computer instructions which are executed and bearing in mind that execution time of each instruction is 2 μsec. Each time a group of instructions, either in the NC or in the AC program, is performed - a software counter (TC) is incremented by the appropriate number. Whenever the counter exceeds the value of 5000, which is equivalent to 10 msec, a new feed-rate calculation takes place. The number 10 msec was arrived at by assuming that the maximum spindle speed which would be used (for steel machining) is 600 rpm and at least 10 samples of measurements would be required per revolution of the spindle.

The ECR starts with the calculating of the cutting force F from the two force signals, F_u and F_v, which are 90 degrees phase-shifted. This

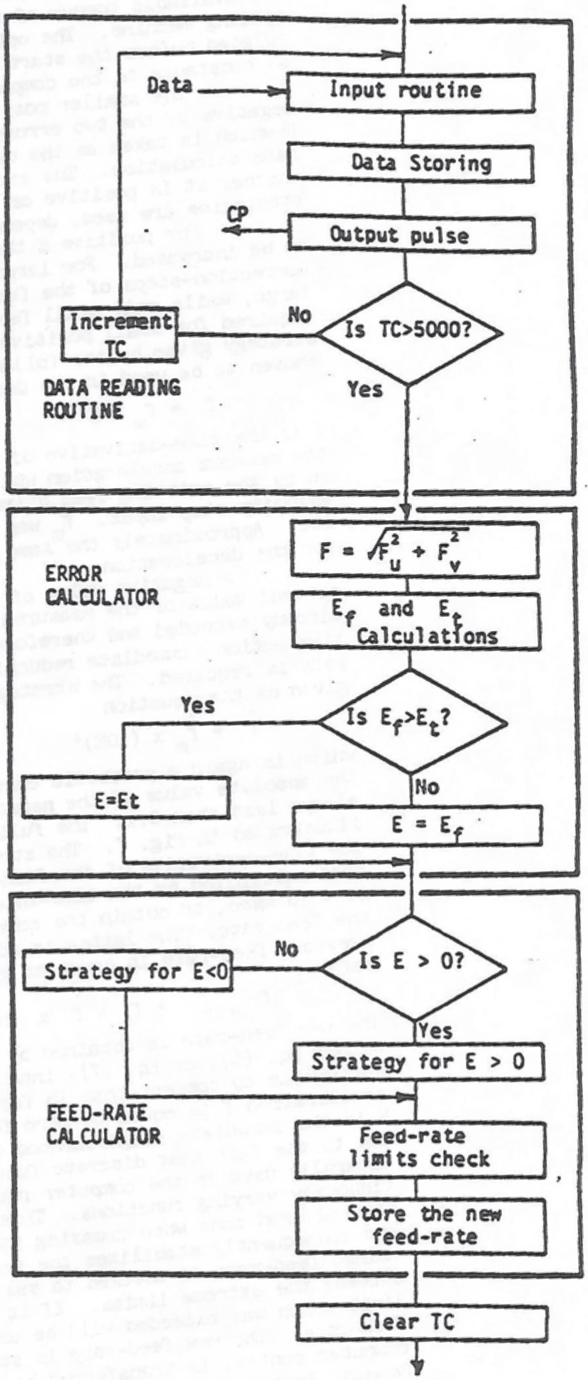


Fig. 6 . The AC Program .

permits the calculation of the force and torque errors, E_f and E_t respectively, which are defined as follows.

$$E_f = 1 - F/F_o \qquad E_t = 1 - T/T_o \tag{5}$$

F_o and T_o are the optimal values of the force and the torque, respectively, and in fact these values are close to the maximum permissible values (around 90 percent). The maximum values depend on the tool geometry and material, as well as on the available torque of the motors that drive the milling machine. The optimal values are calculated before the start of the work and are set as constants to the computer.

The smaller positive or greater negative of the two errors is the critical error E which is taken as the basis for the new feed-rate calculation. The error E is checked to see whether it is positive or negative. Two separate strategies are used, depending on the sign of E.

For positive E the feed-rate will have to be increased. For large positive errors the correction-steps of the feed-rate should be large, while only small feed-rate changes are required for small positive errors. A parabolic strategy given by the following equation was chosen to be used in the described system.

$$\dot{f} = \dot{f}_m \times E^2 \tag{6}$$

$\dot{f}$ is the time-derivative of the feed-rate, $\dot{f}_m$ is the maximum acceleration which can be accomplished by the existing feed drives in response to a velocity step input. $\dot{f}_m$ was measured as 750 ipm/sec. Approximately the same value was found also for the deceleration.

A negative value of E means that the optimal value of the measured variable has been already exceeded and therefore an urgent corrective action (immediate reduction) of the feed-rate is required. The strategy chosen here is given by the equation

$$\dot{f} = \dot{f}_m \times (10E)^2 \tag{7}$$

which is again a parabolic curve. We assume that the absolute value of the negative error is always less than 0.1. The full strategy curve is illustrated in Fig. 7. The strategy calculates the time-derivative of the feed-rate, which is then multiplied by the time-interval Δt, where Δt = 10 msec, to obtain the correction-step of the feed-rate. The latter is added to the previous feed-rate in order to get the new feed-rate

$$f_{(t+\Delta t)} = f_t + \dot{f} \times \Delta t \tag{8}$$

The new feed-rate is obtained by substituting either Eq. (6), or Eq. (7), into Eq. (8). It is worthwhile to comment that in fact the acceleration $\dot{f}$ is equal to zero for the values of E in the immediate neighbourhood of E=0. This is due to the fact that discrete functions are naturally used in the computer rather than continuously varying functions. This creates a natural dead zone when crossing the point E=0 and consequently stabilizes the system. The obtained feed-rate is checked to see whether it exceeds the extreme limits. If it does, the limit which was exceeded will be used as the new feed-rate. The new feed-rate is stored and the computer control is transferred to the Data Reading Routine to receive a new input.

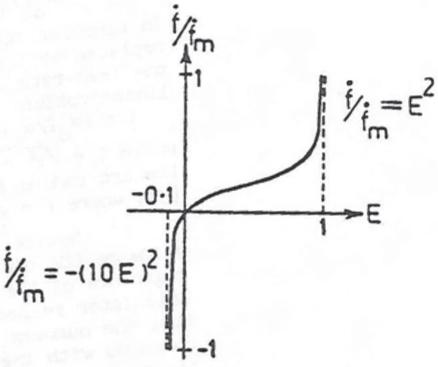


Fig. 7. The change of feed-rate.