OPTIMIZATION OF EXECUTION SPEED OF THE CNC PARAMETRIC PART PROGRAMS

Grzegorz Nikiel

Summary
In the last years were published many works onto subject of the machining operations carried out on the CNC machines. However, many problems exists which are not analysed in them. Especially the optimum CNC programs planning from viewpoint of its analysis and execution on the controller. It has important meaning first of all for the parametric part programs where the complex calculations for determination of tool path are necessary. Discussed experimental results showed that a structure of part program has large influence onto its execution speed. An example of the CNC turning operation of non-axisymmetrical part was presented. For this part an optimization procedure for CNC part program execution speed maximizing was considered.

Keywords: CNC machine tools, optimization, parametric programming

1. Introduction
The machining operations on cutting machines of all types are the subject of many research oriented onto optimization. The researchers mainly deal with a structural as well as a parametrical optimization and their results are described in numerous works. The machining operation carried out on CNC cutting machines are characterized by such features as the many cuts (and cutters), a simultaneity of cuts, high cutting parameters, high surface quality, high
dimensional accuracy, complexity of shape of the machined surface etc. A parametrical optimization of such operations is performed mainly to obtain a low cost and machining time. High productivity of the CNC machines is dependent on many factors in this also a quality of part programs. If the part program is designed in a CAD/CAM environment then is possible optimizing of cutting time, length of tool path, number of used tools etc. (e.g. [1-6]).

Nevertheless, an optimization of one feature is necessary still – the part programs execution speed (practically speed of its interpretation by a CNC controller). In typical machining operations it is not limitation, it does not have large influence onto the final results. Then the time of execution for single block is considerably longer than the time of its analysis and interpretation. However, the time of block analysis is very important in such case as a High Speed Machining which is widely used in the free-form surfaces machining [7]. The speed of machining depends then mainly on internal structure of CNC controller and on quality of interpolation algorithms [8, 9].

Majority of the present CNC programs is non-parametric, with a simple structure. This are first of all the part programs generated automatically by the postprocessors in the CAD/CAM applications. Many of blocks of such programs describe only movement of cutter (an interpolation function + the co-ordinates) and their analysis does not demand large computational efficiency of a CNC controller. It is differently in case of parametric programs (e.g. for machining of complex surfaces) where in many blocks the additional calculations are executed. Necessary time for the co-ordinates calculation can be longer in comparison to a time of tool movement, especially for low efficient CNC controllers. From this cause it can result non-fluent movement of the tool and in consequence – lower quality of machined surface.

In this situation such design of a CNC program is necessary where the time of block execution must be minimized. This time is dependent on two factors: an quality of used algorithm and method of its implementation. In opinion of author more important is first factor. In the second part of this paper an example of analysis is discussed where shortest time of calculations is achieved. This example can be basis for formulating of the general approach in similar cases.

However, in first part the experimental research is described. The dependences between a part program structure and time of its execution was their main aim. Because in typical CNC programs the time of block execution is very short the format of blocks is not exactly analysed. It is differently for example in case of PLC control programs. The PLC controller working in real time demands short time of scan cycle. Therefore several variants of PLC program with use of various instructions are considered [10, 11]. For CNC programs similar approach in practice is infrequently applied [12, 13]. Moreover, the manufacturers of CNC controllers do not give any information about time of execution for the separate functions. In the described experiments this problem was analysed also.
2. **Influence of a part program structure onto its execution speed**

A most optimum method for implementation of algorithm in the CNC part program was main purpose of investigations. Because the Siemens controllers were often used in industrial practice therefore a Sinumerik 810D model was used in the experiments. However, the conclusions from them can be used in case of other controllers.

The following problems were analysed during investigations [14]:

- The speed of calculations on the integer variables – represented by the global R-parameters and by the Local User Data (LUD, pl. Lokalne Zmienne Użytkownika).
- The speed of calculations on the real variables – represented by the global R-parameters and by the Local User Data.
- The speed of calculations for complex functions on the example of square function ($x^2$).
- The speed of calculations for complex arithmetic expressions – in a one block and in the several blocks with decomposition onto simple expressions (estimation of the program execution with use of the heap of processor).
- The speed of calculations with use of the various structural instructions – WHILE … ENDWHILE, REPEAT … UNTIL, FOR… ENDFOR, IF… GOTO/GOTOF.
- The speed of calculations with use of the subprograms.

For accomplishment of the experiments and their results analysis a special part program was written. In this program does not have executed movements of a cutting tool (and other activities of a cutting machine) – a time of calculations is measured only. This time was measured by internal timer (the system variable $\$AC\_TIMER$ [14]). The experiments were executed on the three controllers for which the similar relations of the results were obtained. Finally, for the one CNC controller five repetitions of measurements was executed and their average value was presented below (Table 1-6).

In the test program the required variables were defined by use of `DEF` directive (suitably for the real and integer variables) [14]:

```
DEF REAL VR1,VR2,VR3,VR4,VR5,VR6,VR7,VR8,VR9,VR10
DEF INT VI1,VI2,VI3,VI4,VI5,VI6,VI7,VI8,VI9,VI10,PP
```

<table>
<thead>
<tr>
<th>Description of variables</th>
<th>LUD</th>
<th>R-parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fragment of test program</td>
<td></td>
<td></td>
</tr>
<tr>
<td>VI1=1</td>
<td></td>
<td>R1=1</td>
</tr>
<tr>
<td>VI2=VI1+1</td>
<td></td>
<td>R2=V1+1</td>
</tr>
<tr>
<td>VI3=VI2+1</td>
<td></td>
<td>R3=V2+1</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td>...</td>
</tr>
<tr>
<td>Average time of calculations</td>
<td>1.91 ms</td>
<td>1.48 ms</td>
</tr>
</tbody>
</table>

Table 1. Estimation of the time of calculations on the integer variables
Table 2. Estimation of the time of calculations on the real variables

<table>
<thead>
<tr>
<th>Description of variables</th>
<th>LUD</th>
<th>R-parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fragment of test program</td>
<td>VR1=1.1</td>
<td>R1=1.1</td>
</tr>
<tr>
<td></td>
<td>VR2=VR1+1.2</td>
<td>R2=R1+1.2</td>
</tr>
<tr>
<td></td>
<td>VR3=VR2+1.3</td>
<td>R3=R2+1.3</td>
</tr>
<tr>
<td></td>
<td>....</td>
<td>....</td>
</tr>
<tr>
<td>Average time of calculations</td>
<td>1.92 ms</td>
<td>1.52 ms</td>
</tr>
</tbody>
</table>

Table 3. Estimation of the time of calculations for square function

<table>
<thead>
<tr>
<th>Description of variables</th>
<th>Y = X²</th>
<th>Y = XxX</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fragment of test program</td>
<td>R1=2.156</td>
<td>R1=2.156</td>
</tr>
<tr>
<td></td>
<td>R2=PO(T(R1))</td>
<td>R2=R1*R1</td>
</tr>
<tr>
<td></td>
<td>....</td>
<td>....</td>
</tr>
<tr>
<td>Average time of calculations</td>
<td>2.07 ms</td>
<td>1.89 ms</td>
</tr>
</tbody>
</table>

Table 4. Estimation of the time of calculations with and without use of heap

<table>
<thead>
<tr>
<th>Description of variables</th>
<th>With use of heap</th>
<th>Without use of heap</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fragment of test program</td>
<td>R17=((R1+R2)+(R3-R4)-(R5/R6)-(R7/R8))/R9)*R10</td>
<td>R11=R1+R2</td>
</tr>
<tr>
<td></td>
<td>R12=R3-R4</td>
<td>R12=R3-R4</td>
</tr>
<tr>
<td></td>
<td>R13=R5/R6</td>
<td>R13=R5/R6</td>
</tr>
<tr>
<td></td>
<td>R14=R7*R8</td>
<td>R14=R7*R8</td>
</tr>
<tr>
<td></td>
<td>R15=R11+R12-R13-R14</td>
<td>R15=R11+R12-R13-R14</td>
</tr>
<tr>
<td></td>
<td>R16=R15/R9</td>
<td>R16=R15/R9</td>
</tr>
<tr>
<td></td>
<td>R17=R16*R10</td>
<td>R17=R16*R10</td>
</tr>
<tr>
<td>Average time of calculations</td>
<td>5.78 ms</td>
<td>14.54 ms</td>
</tr>
</tbody>
</table>

The results of measurements (referred to a single computational operation) are presented in Table 1-6. Moreover, an influence of additional elements in the part program structure (e.g. numeration of the blocks, the comments etc.) onto its execution speed was analysed.

Carried out investigations permit to formulate the following conclusions (for the Siemens family controllers) – for faster executing of the CNC part programs the following rules should be accepted:

• to apply calculations on the R-parameters for which speed of calculations for different variables (integer, real) is approximate,

• to avoid applying of the conditional jump instruction (IF ... GOTOx) in favour of the other structural instructions (WHILE … DO, REPEAT … UNTIL etc.),

• to avoid applying of the subprograms,
• to apply minimum number of blocks, the complex arithmetical expressions written in one block,
• to avoid applying of the blocks numeration, comments etc., short blocks with minimum number of words.

Table 5. Estimation of the time of calculations with use of structural instructions

<table>
<thead>
<tr>
<th>Instructions</th>
<th>Fragment of test program</th>
<th>Average time of loop</th>
</tr>
</thead>
<tbody>
<tr>
<td>IF...GOTOx</td>
<td>IF R1&lt;1000 GOTOB N10 R1=0 R2=R1+1 R3=R2+1 ...</td>
<td>18,16 ms</td>
</tr>
<tr>
<td></td>
<td>R1=0 WHILE R1&lt;1000 R2=R1+1 R3=R2+1 ... ENDWHILE</td>
<td>17,80 ms</td>
</tr>
<tr>
<td></td>
<td>R1=0 REPEAT R2=R1+1 R3=R2+1 ... UNTIL R1=1000</td>
<td>18,12 ms</td>
</tr>
<tr>
<td></td>
<td>FOR PP=1 TO 1000 CALL &quot;TEST0&quot; R1=1 R2=R1+1 R3=R2+1 ... ENDFOR</td>
<td>18,04 ms</td>
</tr>
</tbody>
</table>

Table 6. Estimation of the time of calculations for the subprograms use

<table>
<thead>
<tr>
<th>Description of variables</th>
<th>With subprogram</th>
<th>Without subprogram</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fragment of test program</td>
<td>FOR PP=1 TO 1000 CALL &quot;TEST0&quot; ENDFOR</td>
<td>FOR PP=1 TO 1000 R1=1 R2=R1+1 R3=R2+1 R4=R3+1 ... ENDFOR</td>
</tr>
<tr>
<td></td>
<td>TEST0.SPF: R1=1 R2=R1+1 R3=R2+1 R4=R3+1 ... ENDFOR</td>
<td></td>
</tr>
<tr>
<td>Average time of calculations</td>
<td>6,90 ms</td>
<td>1,48 ms</td>
</tr>
</tbody>
</table>
3. An example of optimization of the part program

As example of algorithm speed optimization the seal rings family machining on a CNC turning machine was considered. During operation the external conical surface of ring is machined (Fig. 1). The symmetry axis of this surface is non-parallel to the symmetry axis of internal hole which is treated as a machining datum surface.

![Fig. 1. A machined seal ring and its basic dimensions](image_url)

For correctly machining of the external conical surface is necessary associating of the linear movement of tool in the Z and X axis as well as the rotary movement of spindle in C axis (Fig. 2). For any point P on the conical surface for which the Cartesian co-ordinates \((X_P, Y_P, Z_P)\) are given the co-ordinates \((X_O, Z_O, C_O)\) in the machining co-ordinate system \(X_OZ_OC_O\) was been possible to calculate also (and vice versa of course). An assumption was accepted that during machining the following co-ordinates will be treated as adjustable:

- the \(Z_O\) co-ordinate (from reason of the constant axial feed);
- the \(C_O\) co-ordinate (form reason of the constant rotary speed of spindle).

Third co-ordinate \((X_O)\) in this case is treated as dependent and it is calculated during machining. In practice it means calculation of the intersection point between the cone surface and the straight line which is perpendicular to Z axis and its Z co-ordinate is \(Z_O = Z_P\), moreover it is inclined in the XY plane at \(C_O\) angle to X axis (Fig. 2). For specific conditions of the machining operation the maximum total time of calculations for one intersection point was established as \(\tau_{\text{max}} = 15\) ms.
Fig. 2. Schema for dependences between the co-ordinates of a point on the cone surface

For achievement of the dependences between co-ordinates an assumption was accepted that in first step a transformation of XYZ Cartesian co-ordinate system would be executed. In this way that Z axis will agree with the axis of symmetry of conical surface (modified co-ordinate system is called as X'Y'Z' – Fig. 3). In this case mentioned transformations consist of the shift in Z axis ($D$ dimension) and the rotation around Y axis ($\phi$ angle) – Fig. 4. Assigned co-ordinates will be transformed to original co-ordinate system by use of reverse transformation. Because generally two points of intersection exist ($P_1$ and $P_2$ – Fig. 3) as a final result is accepted closer point to the recently programmed tool position.

Fig. 3. Schema for calculation of the points intersection ($P_1$, $P_2$ – points of intersection, $N_s$ – normal vector of cone, $N_p$ – normal vector of axis)
A general form of a conical elliptic surface equation in the Cartesian co-ordinate system was accepted as follows [15, 16]:

\[
\frac{X^2}{a^2} + \frac{Y^2}{b^2} + \frac{Z^2}{c^2} = 0 \quad (1)
\]

where: \(a, b, c\) – the constant values \((a, b, c > 0)\).

In this situation a equation of conical surface were been possible to express in form:

\[
X^2 + Y^2 + A_S Z^2 + B_S Z + C_S = 0 \quad (2)
\]

where: \(A_S, B_S, C_S\) – the constant values calculated from the given co-ordinates of \(S\) point, radius \(R\) and angle \(\phi\) of cone (Fig. 3):

\[
\begin{align*}
A_S &= -\tan^2 \phi \\
B_S &= 2(R \tan \phi + Z_S \tan^2 \phi) \\
C_S &= -(R^2 + 2Z_S R \tan \phi + Z_S^2 \tan^2 \phi)
\end{align*}
\quad (3)
\]

Moreover, a equation of axis was accepted in the parametric form as follows [1]:

![Transformation of co-ordinate system for the calculation of intersection point](image)
where: $X_p$, $Y_p$, $Z_p$ – the co-ordinates of the given $P$ point of straight line (Fig. 3); 
$\cos \alpha$, $\cos \beta$, $\cos \gamma$ – the direction cosines of straight line; $t$ – the parameter variable, $t \in (-\infty, \infty)$.

In the next step the co-ordinates of points of intersection ($P_1$ and $P_2$) was determined by using a square equation in the following form:

$$w_1t^2 + w_2t + w_3 = 0$$

where: $w_1$, $w_2$, $w_3$ – the coefficients calculated on basis given formulas (2), (4):

$$w_1 = \cos^2 \alpha + \cos^2 \beta + A_3 \cos^2 \gamma$$
$$w_2 = 2X_p \cos \alpha + 2Y_p \cos \beta + (2A_3Z_p + B) \cos \gamma$$
$$w_3 = X_p^2 + Y_p^2 + A_3Z_p^2 + BZ_p + C_3$$

Using described above dependences was designed and tested four variants of the CNC part program:

1. A full version of algorithm with use of the LUD variables (an essential fragment on program listing below). For this variant of algorithm the time of calculations is equal $\tau = 49$ ms $> \tau_{\text{max}}$.

```
LB=DK*ZI ; C_O
; STRAIGHT LINE
XP=0 YP=0 ZP=ZB+DP ; P
CPA=COS(LB) CPB=SIN(LB) CPC=0 ; N_P
; STRAIGHT LINE AFTER ROTATION
XPP=(XP*COS(_FI))- (ZP*SIN(_FI))
YPP=YP
ZPP=(XP*SIN(_FI))+(ZP*COS(_FI))
CPAP=(CPA*COS(_FI))- (CPC*SIN(_FI))
CPBP=CPB
CPCP=(CPA*SIN(_FI))+(CPC*COS(_FI))
; SQUARE EQUATION
_W1=POT(CPAP)+POT(CPBP)+(_A*S*POT(CPCP))
_W2=(2*XP*CPAP)+(2*YP*CPBP)+(2*_A*S*ZP*CPCP)+(B*S*CPCP)
_W3=POT(XPP)+POT(YPP)+(_A*S*POT(ZPP))+(B*S*ZPP)+CS
```
DELTA = POT(_W2) - (4 * _W1 * _W3)
DELTA = SQRT(DELTA)
_T1 = (- _W2 - DELTA) / (2 * _W1)
_T2 = (- _W2 + DELTA) / (2 * _W1)

2. A full version of algorithm with use the R-parameters – on basis of conclusions from chapter 0. Time of calculations in this case is \( \tau = 26 \text{ ms} > \tau_{\text{max}} \).

3. A full version of algorithm with use the R-parameters, with minimization of number of blocks, without the comments and other unnecessary elements of program structure (an essential fragment on listing below) – on basis of conclusions from chapter 0. For this variant of algorithm the time of calculations is equal \( \tau = 16 \text{ ms} > \tau_{\text{max}} \).

4. A simplified algorithm. Because presented above methods do not give desirable time of calculations the another approach was used. During machining the spindle moves step-by-step around C axis about \( \Delta C \) angle (Fig. 5). For the \( N \) sections \( (N = 360^\circ / \Delta C) \) is possible to define the values of \( R_{\text{min}} \) and \( R_{\text{max}} \) radius. Then the \( X \) co-ordinate of any P point is calculated on basis the Tales theorem:

\[
R_p = \frac{Z_p}{G} \left( R_{\text{max}} - R_{\text{min}} \right) + R_{\text{min}}
\]  

(7)

The values of \( R_{\text{min}} \) and \( R_{\text{max}} \) radius for all considered sections are determined by use of described above dependences (1)-(6) before machining is executed and they are saved in register of R-parameters. This procedure can be designed as an additional part program (started only one time) or as a subprogram started at every execution of main part program. Another solution is introduced in [17] where a special application was applied on PC computer.
result of work of this application – the non-parametric part program – is sent to CNC controller by serial interface. During machining the $R_{\text{min}}$ and $R_{\text{max}}$ values of radius are taken from the register of R-parameters and used in main program – below is presented an essential fragment of part program.

\begin{verbatim}
R70=... ; C_O
R34=... ; N
R40=((R70/360)*R34)+300
R41=R40+100
R50=R[R40] ; RMAX FROM TABLE
R51=R[R41] ; RMIN FROM TABLE
R30=(R45* (R50-R51)/R4)+R51 ; XP
\end{verbatim}

For this variant of algorithm the time of calculations is equal $\tau = 9 \text{ ms} < \tau_{\text{max}}$ what finishes the optimization procedure.

4. Concluding remarks

In the present work the questions which are not widely met during planning of the CNC programs were described. For most of the present controllers an execution speed is sufficient in typical cases of machining operations. Then is not necessary saving-up on the part program structure (e.g. by absence of block numeration, comments etc.). Nevertheless large importance for the machining operation and its effects has speed of analysis and execution of part programs, especially in case of controllers with low computational efficiencies. Similarly how in described example a machining operation can be impossible or its results can be unacceptable. Just the optimized part program made possible a machining operation. In addition an area of changes for the adjustable cutting parameters was obtained. Moreover, the paper showed dependence between a CNC part program structure and speed of its analysis. Study of this dependence is very
important because such information usually is not included in the documentation of manufacturers.

References


Received in June 2009