The International Journal of Advanced Manufacturing Technology (2024) 130:4635–4648
https://doi.org/10.1007/s00170-024-12960-x

ORIGINAL ARTICLE

Design and implementation of an innovative canned cycle for variable pitch thread cutting on CNC milling machines

Sotiris L. Omirou[1] · Charalambos A. Chasos[1]

Received: 27 July 2023 / Accepted: 30 December 2023 / Published online: 18 January 2024

Abstract

This paper presents the development of a new canned cycle for CNC milling machines capable of cutting threads with variable pitch, addressing a significant limitation in modern CNC systems. While existing CNC milling machines provide canned cycles for thread cutting, they lack the capability to program variable pitch, necessitating reliance on CAM systems. In response, the proposed canned cycle offers increased flexibility, quicker setup times, and reduced dependence on CAM systems. The cycle accommodates both internal and external threads, supports left-hand or right-hand threading, and incorporates a user-friendly control panel for easy programming. To validate its efficiency, the canned cycle is implemented using a G-code parametric algorithm, and a series of simulation tests are conducted. The results demonstrate the viability and advantages of the new CNC milling feature in cutting threads with variable pitch, opening up possibilities for various mechanical applications.

Keywords CNC machining · Canned cycles · Milling · Interpolation algorithms

1 Introduction

Canned cycles in CNC machines are pre-programmed sequences of commands that simplify and automate specific machining operations [1–3]. Their importance is acknowledged since they reduce the complexity of manual programming, and improve operational efficiency. Canned cycles provide a range of predefined commands customized to various machining tasks, such as pockets, slots, drilling, and more. For example, a pocket canned cycle enables the efficient machining of rectangular or circular pockets with defined parameters like dimensions, depths, and feedrates. By utilizing canned cycles, machinists can easily and accurately execute common machining operations, saving time and ensuring consistent results.

Thread cutting is a commonly performed machining operation across various machine tools, and thread milling provides a highly valuable option for numerous applications, serving as an advantageous alternative to thread forming [4]. Among CNC machines, drill tap centers, turning centers, and milling centers are the primary choices for machinists, depending on their availability within the manufacturing unit. To streamline the threading process, CNC machines offer specialized thread canned cycles to programmers. By utilizing these canned cycles, users can efficiently set up the operation by inputting the required values for parameters such as feedrate, workpiece's radius, pitch, and depth into the control system. When it comes to CNC milling machines, nearly all types offer one or more canned cycles specifically designed for thread cutting. However, a thorough examination of the instruction sets within leading CNC milling systems exposes a notable limitation. Despite the inherent versatility and functionality of CNC milling machines, they currently lack the capability to program variable pitch threads using canned cycles, thus requiring reliance on CAM systems.

In precise engineering and manufacturing fields, the variable pitch threads are required for various applications, including components or parts for automotive vehicle engines, aerospace systems' fasteners, and medical implants [5–8]. The various applications of variable pitch threads reveal the imposed need for simplified and easy programming in order to generate variable pitch threads. However, traditional CNC systems pose programming limitations which result in increased manufacturing costs. The present study addresses pertinent challenges in manufacturing with the introduction of a novel method which attempts to innovate in the machining of variable pitch threads. Thus, limitation of conventional CNC systems can be surpassed, while with the new method it

Sotiris L. Omirou
eng.os@frederick.ac.cy

[1] Department of Mechanical Engineering, Frederick University, Nicosia, Cyprus

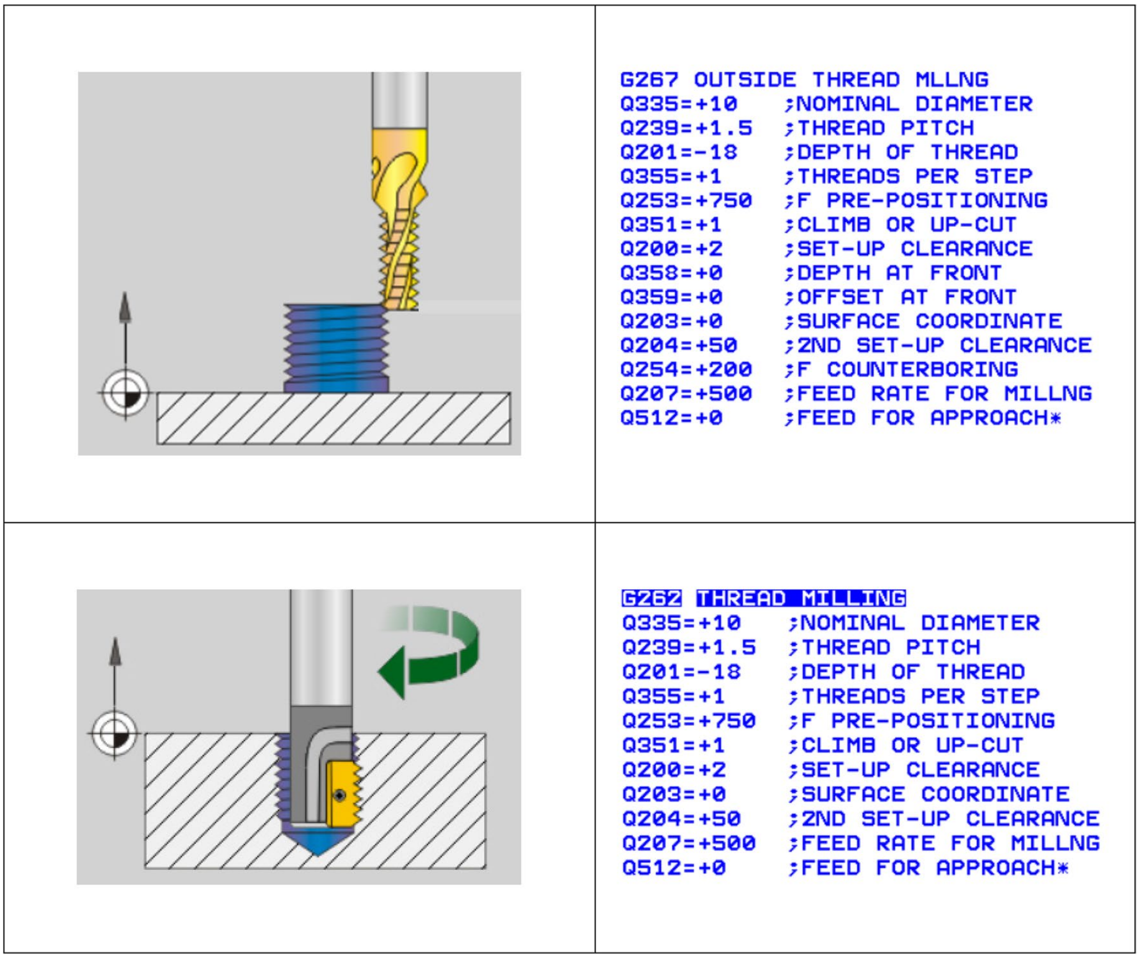

Fig. 1 G262 / G267 – Internal / External thread milling canned cycles of Heidenhain control

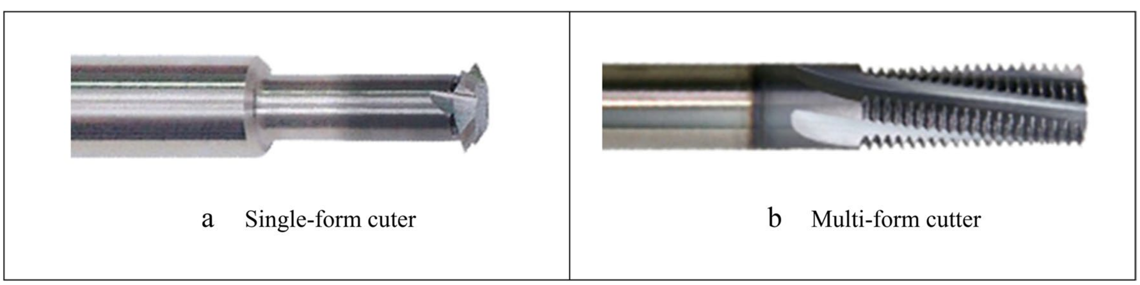

Fig. 2 Basic types of thread milling cutters

is envisaged that the manufacturing process will be optimized, as well as new options for flexible and improved designs and optimal products can be achieved.

The present paper presents a creative canned cycle for milling variable pitch threads. The work presented is aligned with the current trends in CNC machining, which are characterized by an expansion in research efforts to create new machining techniques suitable for specific requirements and capabilities with enhanced features. The main idea of this new canned cycle is based on a G-code parametric interpolation algorithm, which is carefully designed to guide the cutting tool along the defined path. The cycle prioritizes two crucial factors, namely ease of use in programming and the ability to provide comprehensive solutions for cases involving variable pitch threads. A user-friendly framework is developed through an intuitive control panel, which allows users to input the necessary data for their specific machining needs. Simultaneously, the system offers flexibility by enabling users to define thread characteristics via the control panel, and by setting precise configuration of parameters, including workpiece radius, initial or final pitch, number of turns, left or right-hand threading, feed rate, and spindle speed.

The rest of the paper is organized as follows: Sect. 2 provides a review of the existing thread milling techniques applied in modern CNC milling controllers. Section 3 describes the platform of parametric programming and the functions it can perform. Section 4 presents the development of the G-code parametric interpolation algorithm and its embodiment under specific G-codification in the control of a CNC milling machine. Section 5 demonstrates the functionality of the new canned cycle through a series of simulation tests. Finally, Sect. 6 summarizes the conclusions of the paper, highlighting the potential advancements in CNC milling and its benefits for various industries requiring threads with variable pitch.

2 Existing thread milling techniques

In this section, the focus is on exploring the thread milling operation provided by widely used commercial CNC milling machine controllers. It examines the programming aspects of these controllers, which offer specific functionalities for both external and internal threading operations.

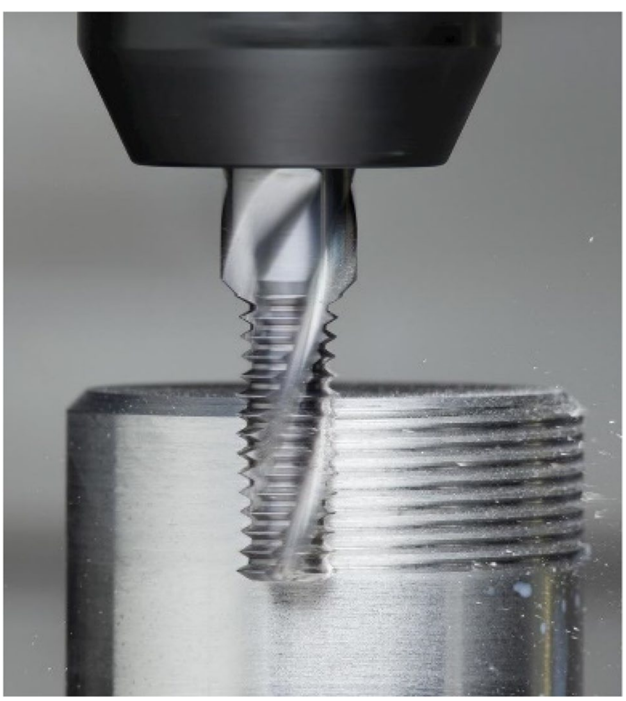

Fig. 3 External thread milling with a multi-form cutter

2.1 Heidenhain control

Heidenhain control system [9] provides a range of parameters denoted as Q, allowing users to program both internal and external threading operations (Fig. 1). By using parameter Q355 with a value of either 0 or 1, users can specify the type of cutter they wish to utilize, whether it is a single form or multi-form thread milling cutter (Fig. 2).

2.2 HAAS control

For HAAS control [10], thread milling involves employing a conventional G02 or G03 command to achieve circular motion in the X–Y plane, followed by a Z-axis command to establish the desired thread pitch. This process completes one revolution of the thread, while the remaining revolutions are generated by the teeth of the multiple-form cutter (Fig. 3).

A typical code line for this operation might appear as follows:

N100 **G**02 **I** – 40.0 **Z** – 1.5**F** 100 (HAAS code)

This line creates a 1.5 mm pitch thread with a 40 mm radius.

2.3 FANUC control

Similarly, for FANUC control [11], thread cutting using an appropriate fitting tool is facilitated through the dedicated G33 code, as shown in Fig. 4.

The corresponding G-code line takes the following format:

N100 **G**33 **Z** – 50. **F**1.5 (FANUC code)

Parameters Z and F represent the hole depth and pitch, respectively, creating a 1.5 mm thread in a 50 mm depth hole.

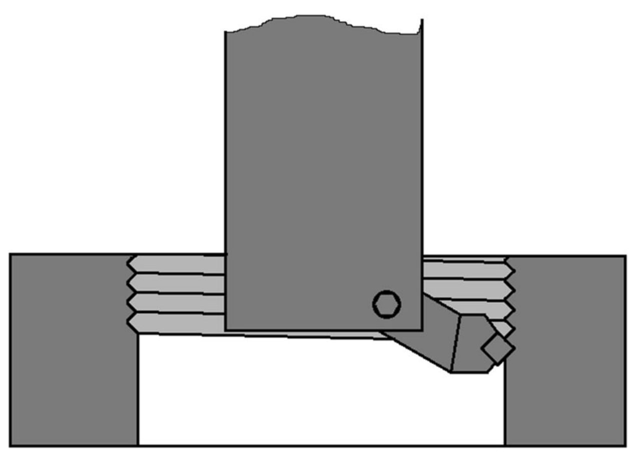

Fig. 4 Internal thread milling with a fitting tool

Despite the convenience of these programming techniques, they share a common limitation—the absence of provisions for threads with variable pitch. Existing programming approaches rely on predefined cycles with fixed pitch values, while cases requiring variable pitch threads often require CAM system interpretation, involving laborious design and calculation processes. In response to this limitation, the present work introduces an innovative canned cycle, allowing straightforward and user-friendly programming of threads with variable pitch. This new cycle is developed using a parametric programming language, present in all modern CNC controls under various names.

3 CNC parametric programming

CNC parametric programming is a versatile platform accessible in widely used controllers, known by different names based on the specific controller, such as Fanuc custom macro B, Fadal macro, Okuma user task, and Heidenhain parametric technique. This platform has been utilized by several research works [12–16] to interpret challenging machining cases.

This section focuses on providing a detailed description of Heidenhain's parametric programming, which is the programming language utilized in this paper. For an in-depth understanding of the language, readers are referred to Lynch's comprehensive work [17]. In the Heidenhain control system, parameters are identified using the letter Q followed by an integer number. These parameters can be assigned numerical values directly or through arithmetic or trigonometric operations. The Q parameters can also be dynamically updated during program execution and continuously checked to determine if they meet specific conditions, thereby influencing the program flow. Apart from the Q parameters, the letter D is used to encode various functions, with specific functions denoted by numbers 1 to 13 (refer to Table 1). To provide a clear understanding of the language's structure, the table includes examples and explanations for each individual D code. It is important to note that Q parameters and fixed numerical values can coexist within the same function, enabling greater flexibility in programming.

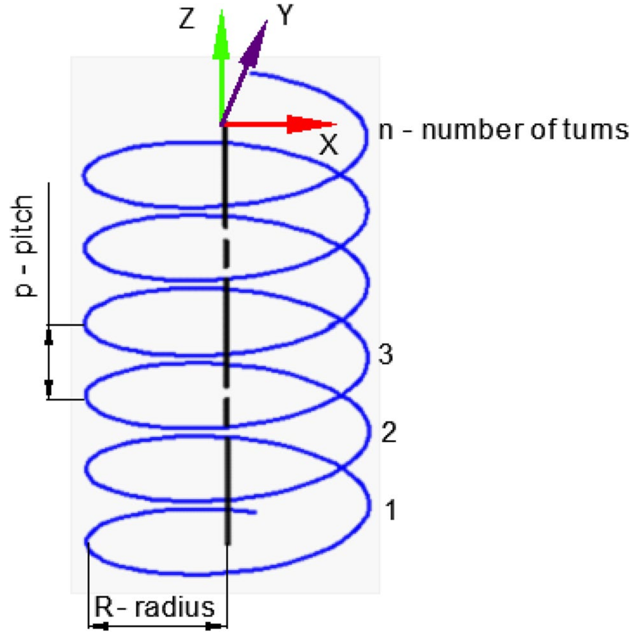

Fig. 5 The elements of a helix

4 Development of the G-code parametric interpolation algorithm

4.1 Design considerations

The development of an advanced, accurate, and user-friendly canned cycle for milling threads with variable pitch required

Table 1 Explanation of parameters employed in the Heidenhain control system

D CODE	FUNCTION	EXAMPLE	EXPLANATION
D00	Assignment	D00 Q1 P01 20	Assignment of the value 20 to parameter Q1
D01	Addition	D01 Q2 P01 Q3 P02 Q4	The result of (Q3 + Q4) is assigned to Q2
D02	Subtraction	D02 Q5 P01 Q6 P02 8	The result of (Q6 – 8) is assigned to Q5
D03	Multiplication	D03 Q8 P01 Q9 P02 -2	The result of (–2)(Q9) is assigned to Q8
D04	Division	D04 Q10 P01 Q11 P02 4	The result of Q11/4 is assigned to Q10
D05	Square root	D05 Q12 P01 Q13	The square root of Q13 is assigned to Q12
D06	Sine	D06 Q14 P01 Q15	The result of sin(Q15) is assigned to Q14
D07	Cosine	D07 Q16 P01 Q17	The result of cos(Q17) is assigned to Q16
D09	If equal, jump	D09 P01 Q20 P02 5 P03 1	If Q20 is equal to 5, jump to label 1
D10	If not equal, jump =	D10 P01 Q21 P02 Q22 P03 2	If Q21 is not equal to Q22, jump to label 2
D11	If greater, jump	D11 P01 Q23 P02 0 P03 3	If Q23 is greater than 0, jump to label 3
D12	If less, jump	D12 P01 Q24 P02 0 P03 4	If Q24 is less than 0, jump to label 4
D13	Calculate angle	D13 Q30 P01 Q25 P02 Q26	The result of arctan(Q25/Q26) is assigned to Q30

careful consideration of several key factors. The algorithm was designed to meet specific performance objectives, ensure ease of use during programming, and offer flexibility for comprehensive coverage of threading cases. **Performance**: The primary objective of the canned cycle was to enable efficient thread cutting with variable pitch on CNC milling machines. To achieve this, the algorithm needed to support both internal and external threads, accommodate left- or right-hand threading, and allow for the selection of variable pitch by defining either the starting or ending pitch alongside the number of turns. The tool step motion needed to be adjustable, allowing machinists to set the desired precision. **User-friendliness**: The canned cycle aimed to accommodate the programming process and minimize the need for complex CAM system interpretations. A dedicated control panel was designed as an interface between the user and the canned cycle. The control panel incorporated all the necessary parameters, simplifying data input and reducing the risk of errors during programming. **Flexibility**: Recognizing the diversity of threading cases in practical applications, the algorithm was designed with the capability to adapt to various cases. The control panel allowed users to input specific values for parameters such as helix radius, starting or ending pitch, number of turns, left-hand or right-hand thread, feedrate, and spindle speed, thereby providing comprehensive flexibility in programming.

4.2 3D Helical motion for milling threads with variable pitch

In order to cut threads with variable pitch on a 3-axis CNC milling machine, the capability of generating 3D motion along a helix with a variable pitch is essential. This necessitates obtaining the solution of parametric equations x(t), y(t), z(t) of the helix within an interpolation algorithm framework. The parametric equations of a helix [18] with constant radius and pitch are given by Eq. 1 (Fig. 5).

$$x(t) = Rsin(360nt)$$
$$y(t) = Rcos(360nt) \quad (1)$$
$$z(t) = npt$$

where,

- R the radius of the helix
- n the number of turns
- p the pitch
- t the variable ϵ[0…1] A dimensionless parameter that defines the accuracy of the generation of helix's shape

Table 2 Summary of Q parameters used by the algorithm and their descriptions

Parameter Q	Description
Q1	0 for left-hand thread, 1 for right-hand thread
Q2	0 for increasing pitch, 1 for decreasing pitch
Q3	Pitch – p (mm)
Q4	R_1+R_2 for external threads R_1-R_2 for internal threads
Q5	Number of turns—n
Q6	Precision step—t
Q7	Feedrate F (mm/min)
Q8	Spindle speed S (rpm)

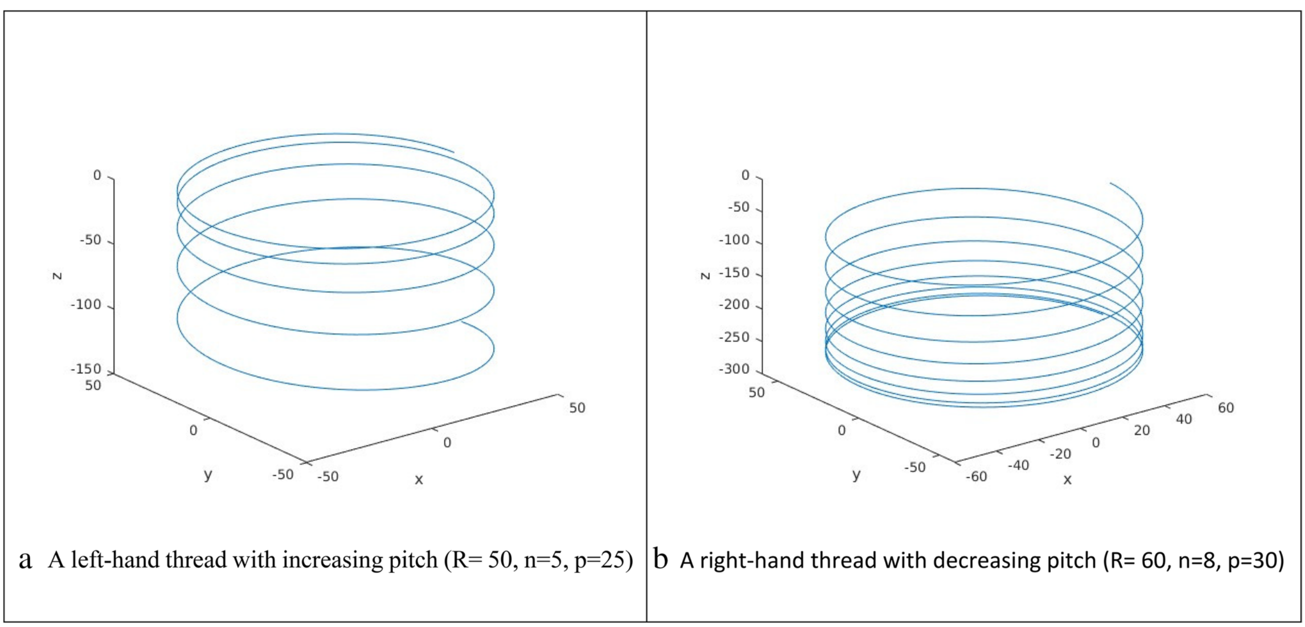

Fig. 6 The two types of thread with variable pitch (**a**. increasing, **b**. decreasing)

'τ' serves a dual purpose in defining the helix's geometry. First, it determines the angular step position for X and Y coordinates.

As 't' varies from 0 to 1, it effectively traces the path around the helix in an angular sense. Second, 't' simultaneously controls the vertical departure from the starting point, which is expressed by the Z coordinate. Therefore, it plays a pivotal role in defining both the planar angular position and the vertical position along the helix. A smaller value of 't' yields shorter linear segments, which, in turn, enhances the precision of the traced helix.

For a helix with a constant radius and variable pitch, the equations (Eq. 1) can be expressed as follows:

$$\begin{aligned} x(t) &= Rsin(360nt) \\ y(t) &= Rcos(360nt) \\ z(t) &= npt^2 \end{aligned} \tag{2}$$

To adapt these equations based on the CNC milling machine's tool-part setup, allowing the tool to move from the top to the bottom of the part with the upper surface considered as Z = 0, the following adjustments are made for the case of an increasing pitch:

$$\begin{aligned} x(t) &= Rsin(360nt) \\ y(t) &= Rcos(360nt) \\ z(t) &= -npt^2 \end{aligned} \tag{3}$$

Accordingly, for the case of a decreasing pitch the equations take the form:

$$\begin{aligned} x(t) &= Rsin(360nt) \\ y(t) &= Rcos(360nt) \\ z(t) &= np(1-t)^2 - np \end{aligned} \tag{4}$$

Furthermore, by exchanging the sine and cosine functions in the first two equations, it is possible to achieve a left-hand or right-hand thread type. The verification of the two cases, a left-hand thread with increasing pitch (R = 50, n = 5, ending

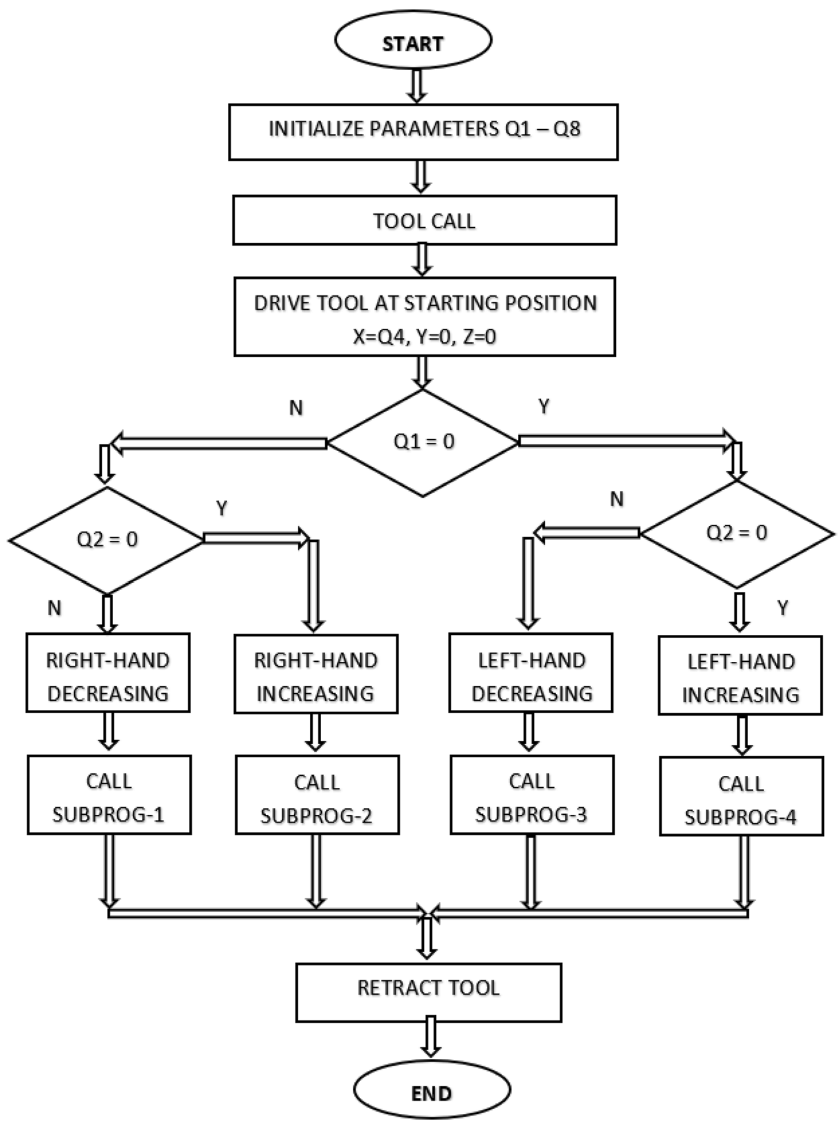

Fig. 7 Flowchart of the main program

pitch p = 25) and a right-hand thread with decreasing pitch (R = 60, n = 8, starting pitch p = 30), is depicted graphically in Fig. 6 through two distinct numerical examples.

4.3 Implementation of the G-code algorithm

The G-code algorithm is formulated within the environment of Heidenhain control. Additionally, it can be easily translated for use with any other CNC control that supports parametric programming. Inputting the required thread data and cutting conditions into the G-code algorithm is achieved through the utilization of Q parameters (see Sect. 3). The algorithm comprises a main program and four subprograms. The main program initializes thread data by setting parameters Q1-Q8 (Table 2), drives the cutter to the starting point, and controls the execution flow towards one of the four subprograms. Each of the four subprograms is specifically designed to handle a distinct case of threads, namely: right-hand decreasing pitch, right-hand increasing pitch, left-hand decreasing pitch, and left-hand increasing pitch. The flowcharts in Figs. 7 and 8 form the basis for developing the G-code parametric algorithm, as they illustrate the logical structure of the main program and the four subprograms. While the flowchart of the subprogram (Fig. 8) represents the case of a right-hand decreasing pitch thread, similar flowcharts can be easily generated using the appropriate equations for the other three cases.

The zero-reference point is positioned at the top center of the helix. Driving the tool to the starting position involves two steps. The first step is common for both external and internal threads, where the tool is directed to the zero-reference point. In the second step, for internal threads, the tool is driven linearly at $X = R_1 - R_2$, $Y = 0$, and $Z = 0$, while for external threads, it moves to $X = R_1 + R_2$, $Y = 0$, and $Z = 0$, (with R_1 representing the helix radius and R_2 the tool radius). The user initializes this value through the Q4 parameter.

Based on the flowcharts presented above, the logic depicted in them facilitated the development of the G-code parametric algorithm. The corresponding code, along with accompanying comments, is provided in Tables 3 and 4. Table 3 presents the main program, while Table 4 details subprogram 1. The remaining three subprograms were developed by utilizing the appropriate equations.

4.4 Format of the new cycle

Generally, canned cycles are standardized and codified using a G-code in conjunction with a series of parameters. It is important to note that CNC controllers intentionally reserve certain G-codes for future use, allowing for customized implementations and increased flexibility. Specifically,

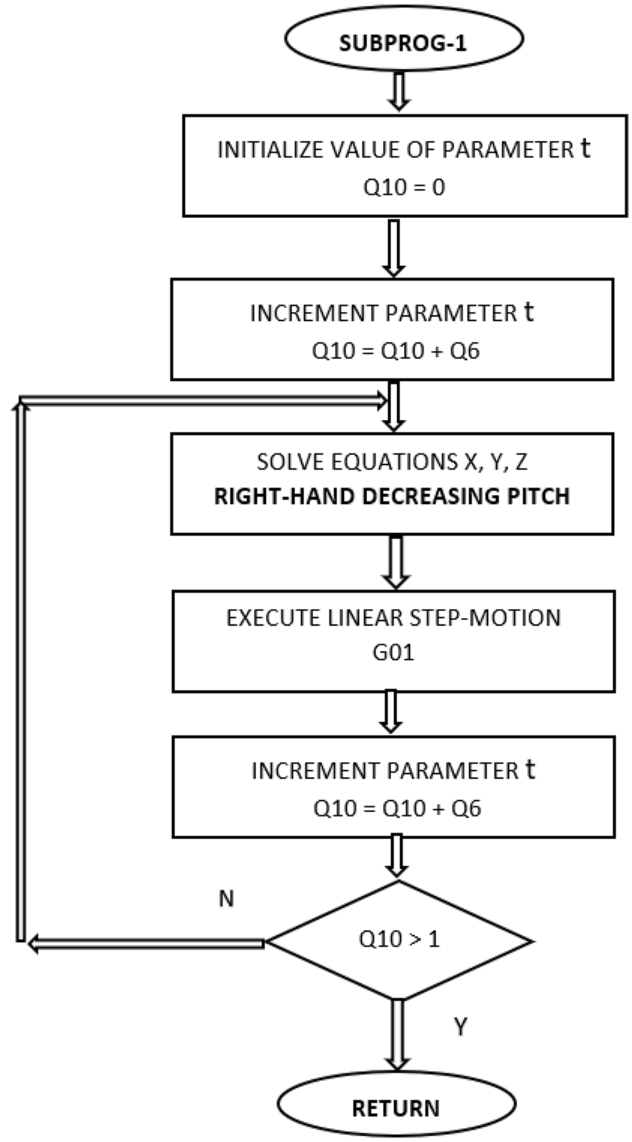

Fig. 8 Flowchart of SUBPROG-1

in the context of Heidenhain control, the unassigned G100 code has been selected for the proposed implementation. Complementary data required for the canned cycle can be expressed through a statement following the format:

G100 P01 xx P02 xx

P03xx P04 xx P05 xx P06 xx,

xx, xx P07xx P08xx

Within this format, the G100 code is suggested to represent the canned cycle for milling threads with variable pitch. The parameters P00 to P08 are utilized to specify the following:

Table 3 List of G-code for the main program

Main program (Heidenhain G-code)	Comments
%THREAD_VAR G71 * N10 D00 Q1 P01 +1* N20 D00 Q2 P01 +1* N30 D00 Q3 P01 +30* N40 D00 Q4 P01 +60* N50 D00 Q5 P01 +8* N60 D00 Q6 P01 +0.001* N70 D00 Q7 P01 +100* N80 D00 Q8 P01 +400* N90 T1 G17 SQ8* N100 M3* N110 G90 G00 X+0 Y+0 Z+20* N120 G01 Z+0 FQ7* N130 D01 Q9 P01 +Q1 P02 +Q2* N140 D09 P01 +Q9 P02 +2 P03 1* N150 D09 P01 +Q9 P02 +0 P03 2* N160 D09 P01 +Q1 P02 +1 P03 3* N170 % SUBPROG_3.I N180 L4* N190 G98 L1* N200 % SUBPROG_1.I N210 L4* N220 G98 L2* N230 % SUBPROG_4.I N240 L4* N250 G98 L3* N260 % SUBPROG_2.I N270 G98 L4* N280 G00 X+0 Y+0* N290 G00 Z+100* N300 M30* N99999999 %THREAD_VAR G71 *	Input thread data and cutting conditions (Initialization of parameters Q1-Q8) Tool call – Start rotation CW Position tool at zero-reference point Check the type of thread and direct the flow to one of the 4 subprograms Subprogram-3: Left-hand decreasing Subprogram-1: Right-hand decreasing Subprogram-4: Left-hand increasing Subprogram-2: Right-hand increasing Retract tool Program end

Table 4 List of G-code for the Subprogram-1

Subprogram-1 (Heidenhain G-code)	Comments
%SUBPROG_1 G71 * N10 D00 Q10 P01 +0* N20 D01 Q10 P01 +Q10 P02 +Q6* N30 G98 L5* N40 D03 Q11 P01 +Q10 P02 +Q5* N50 D03 Q12 P01 +Q11 P02 +360* N60 D06 Q13 P01 +Q12* N70 D07 Q14 P01 +Q12* N80 D03 Q30 P01 +Q13 P02 +Q4* N90 D03 Q31 P01 +Q14 P02 +Q4* N100 D03 Q15 P01 +Q3 P02 +Q5* N110 D02 Q16 P01 +1 P02 +Q10* N120 D03 Q17 P01 +Q16 P02 +Q16* N130 D03 Q18 P01 +Q15 P02 +Q17* N140 D02 Q32 P01 +Q18 P02 +Q15* N150 G01 X+Q30 Y+Q31 Z+Q32* N160 D01 Q10 P01 +Q10 P02 +Q6* N170 D12 P01 +Q10 P02 +1 P03 5* N99999999 %SUBPROG_1 G71 *	Update parameter t. Set a label. Solve equations for **right-hand decreasing pitch**. Execute a 3D step linear motion. Check the end condition. Repeat the cycle or return to the main program

P01 – 0 for left – hand thread, 1 for right – hand thread
P02 – 0 for increasing pitch, 1 for decreasing pitch
P03 – the pitch(mm)
P04 – $R_1 + R_2$ for external threads, $R_1 - R_2$ for internal threads
P05 – the number of turns
P06 – step for parameter t
P07 – Feed rate F(mm/min)
P08 – spindle speed S(rpm)

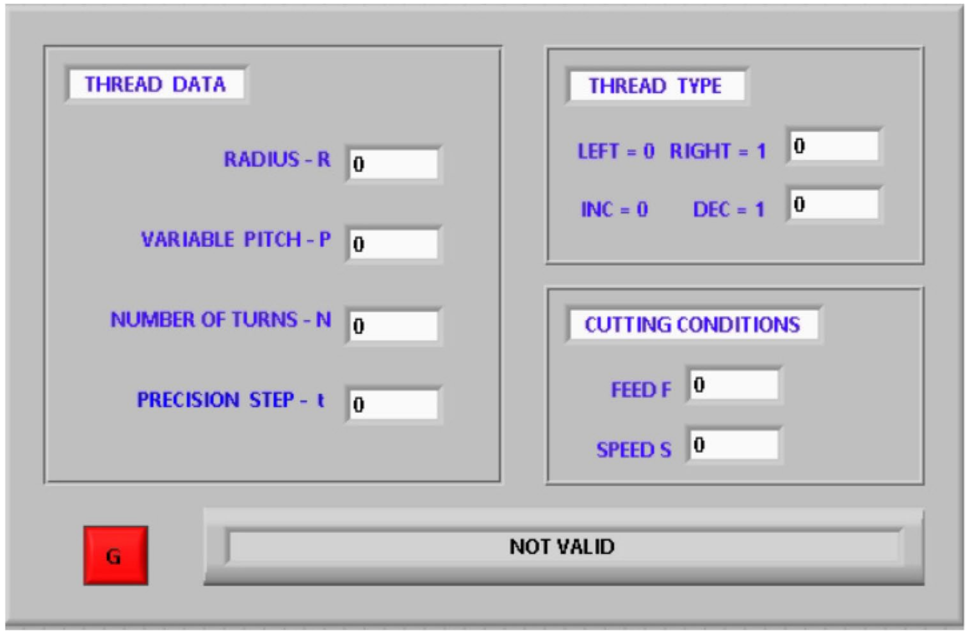

Fig. 9 Control panel for variable pitch thread milling

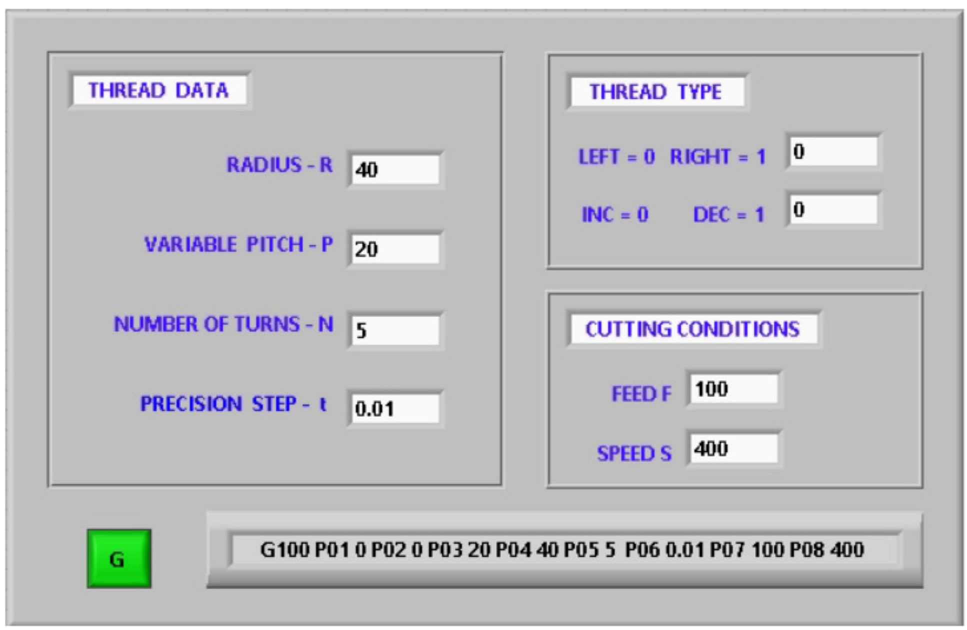

Fig. 10 Control panel of Test-1

In all cases, the appropriate tool for threading is a single-form cutter (Fig. 2a).

4.5 The control panel

To ensure a user-friendly environment and prevent the entry of unacceptable values, a control panel (Fig. 9) has been designed using a high-level programming language. This panel incorporates all the necessary parameters that need to be updated for any new desired case of variable pitch thread cutting. By appropriately completing the designated fields and pressing the "G" button, the corresponding values are transferred to the respective parameters P01…P08 in the G100 statement. The completed statement is then displayed on the screen, ready to be transferred to the CNC controller. In addition to its user-friendly interface, the panel includes a built-in validation mechanism that checks the inserted values for acceptability. Only when all the values are valid does the G100 statement is displayed ready for further processing.

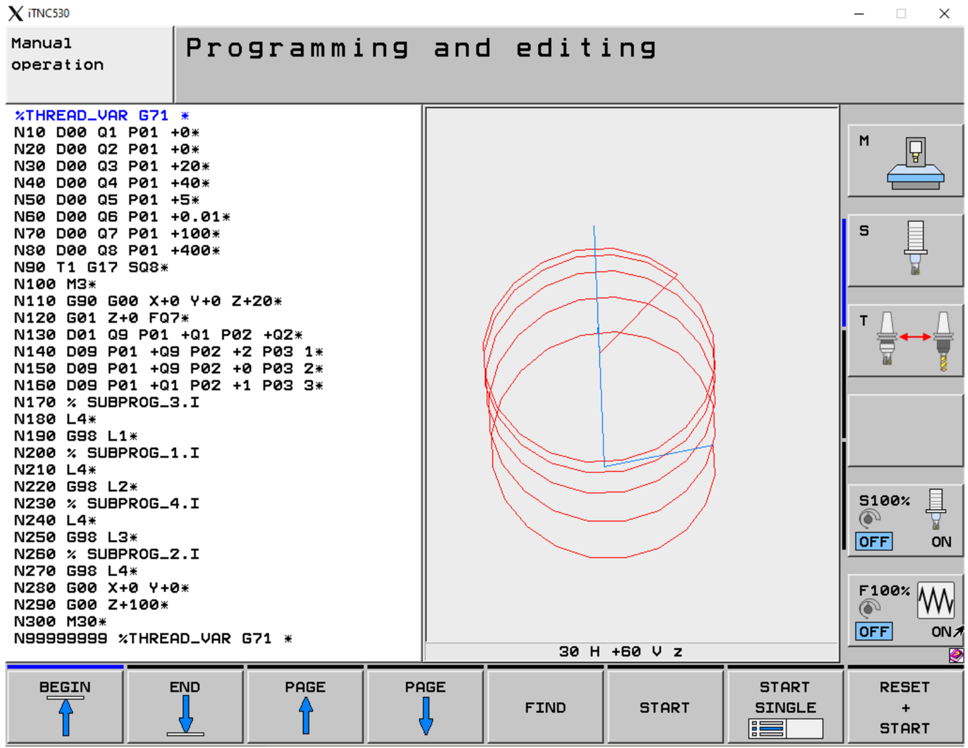

Fig. 11 Simulation result of Test-1: Left-hand increasing pitch (R = 40, p = 20, n = 5, t = 0.01, F = 100, S = 400)

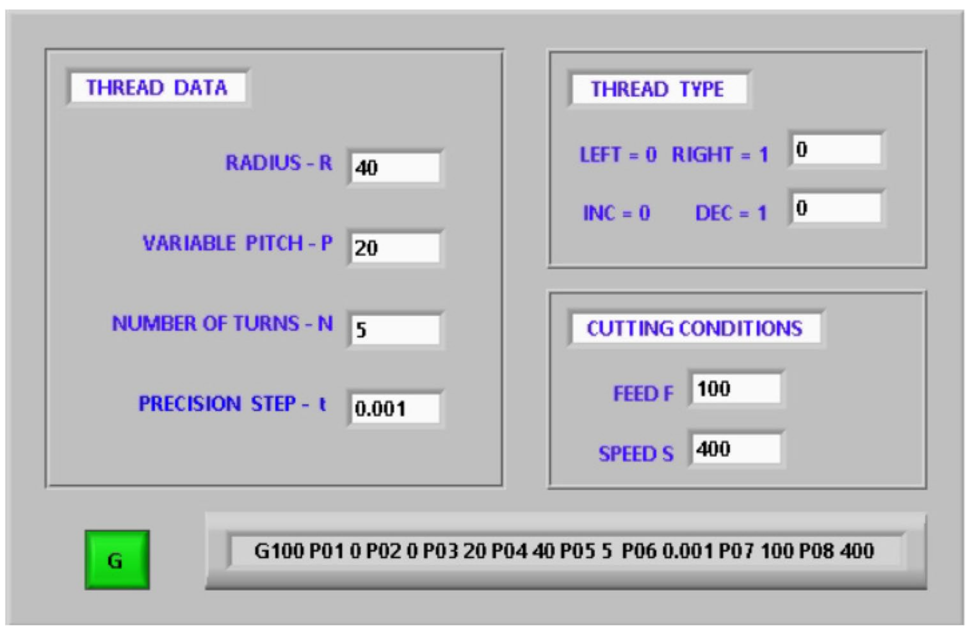

Fig. 12 Control panel of Test-2

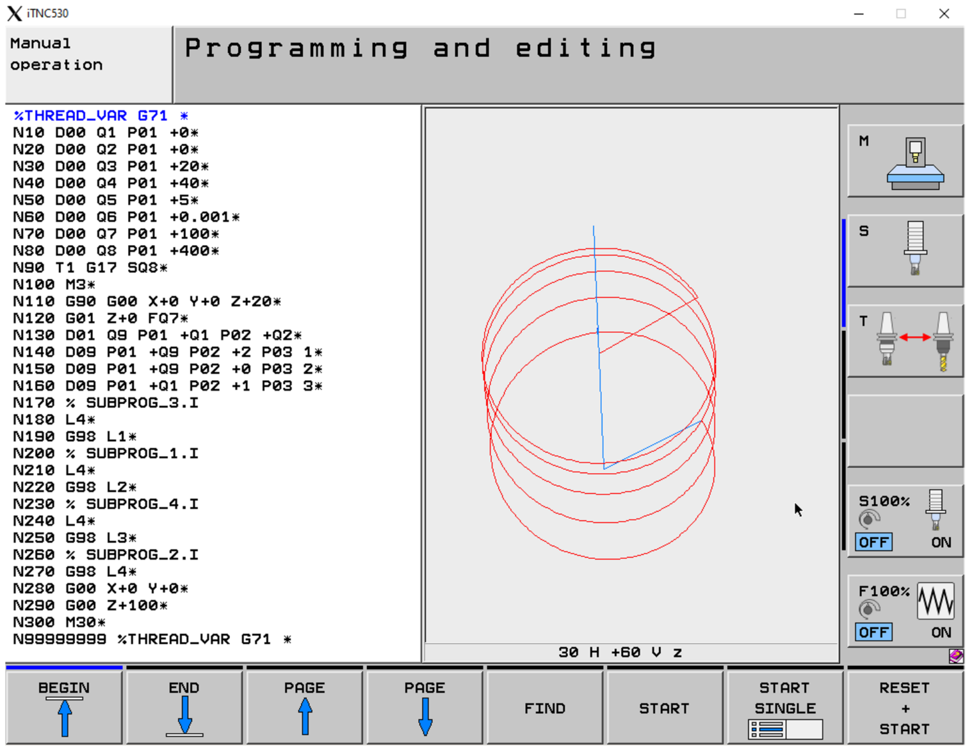

Fig. 13 Simulation result of Test-2: Left-hand increasing pitch ($p=20$, $R=40$, $n=5$, $t=0.001$, $F=100$, $S=400$)

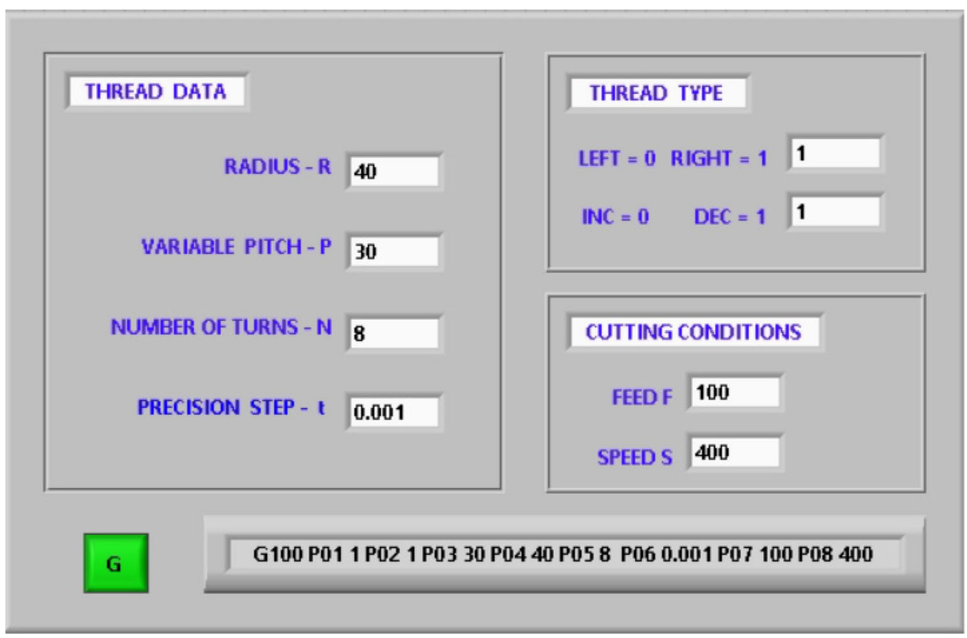

Fig. 14 Control panel of Test-3

5 Simulation tests

The simulation tests (Figs. 10, 11, 12, 13, 14, 15, 16, and 17) demonstrate the performance of the proposed canned cycle for different cases of variable pitch thread cutting on CNC milling machines. TEST-1 and TEST-2 focus on the left-hand increasing pitch case with variations in precision ($t=0.01$ and $t=0.001$), which is a useful option when performing roughing and finishing operations. On the other hand, TEST-3 and TEST-4 explore the right-hand decreasing pitch case, showcasing the cycle's adaptability to varying radii ($R=40$mm, $R=60$mm), pitch (30mm, 25mm), and turns (8 and 12). Each simulation test is associated with its respective and appropriate update of the control panel, describing the corresponding threading case.

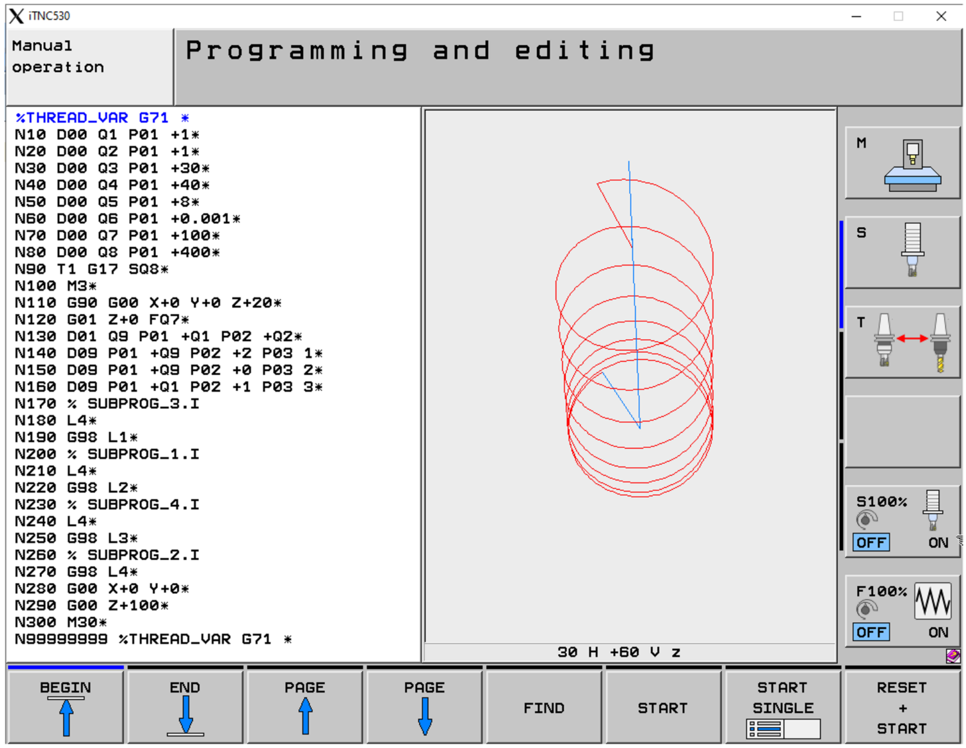

Fig. 15 TEST-3: Simulation result of Test-3: Right-hand decreasing pitch ($p=30$, R = 40, $n=8$, t = 0.001, F = 100, S = 400)

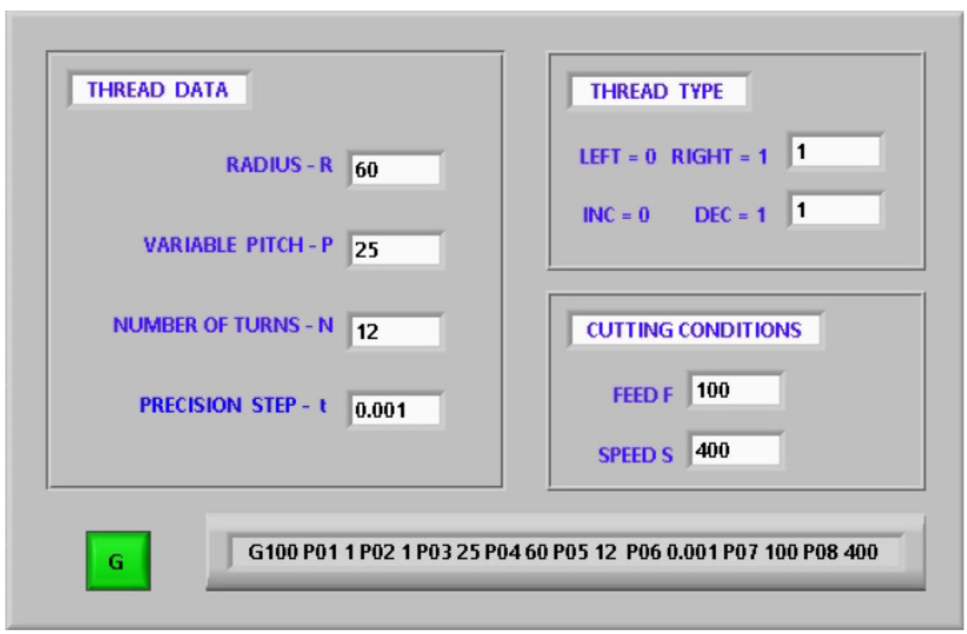

Fig. 16 Control panel of Test-4

TEST-1: Left-Hand Increasing Pitch (Precision t = 0.01).

TEST-2: Left-Hand Increasing Pitch (Precision t = 0.001).

TEST-3: Right-Hand Decreasing Pitch (R = 40mm, P = 30mm, n = 8, t = 0.001).

TEST-4: Right-Hand Decreasing Pitch (R = 60mm, P = 25mm, n = 12, t = 0.001).

Upon analysis and consideration of the presented tests, it should be clarified that the equations provided in the manuscript (2, 3, and 4) offer CNC operators the option to select, for a specific length, both the number of turns and the value of the variable pitch. The condition is that the product of the selected number of turns and the chosen pitch value should equal the desired length. Consequently, operators can judge

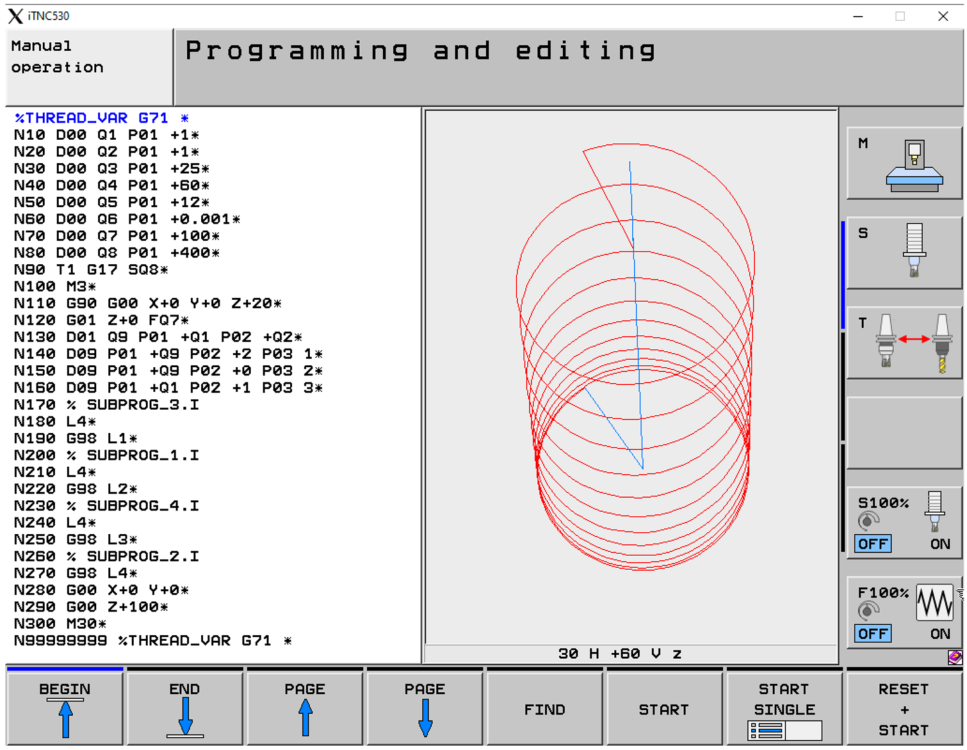

Fig. 17 TEST-4: Simulation result of Test-4: Right-hand decreasing pitch ($p = 25$, R = 60, $n = 12$, t = 0.001, F = 100, S = 400)

and decide on these values depending on the application. Notably, the operator can freely choose the radius, precision step 't,' and the appropriate cutting conditions (F and S) within the scope of the respective case.

6 Conclusions

This paper introduces an innovative canned cycle for CNC milling machines, allowing cutting threads with variable pitch. Modern CNC milling systems lack the capability to program variable pitch threads using canned cycles, relying on CAM systems. The proposed canned cycle offers increased flexibility, quicker setup times, and reduced reliance on CAM systems. It accommodates both internal and external threads, supports left-hand or right-hand threading, and provides a user-friendly interface for programming. The control panel interface simplifies data input, while simulation tests confirm the cycle's efficiency. The new canned cycle presents a potential advancement in CNC milling, providing valuable benefits for various industries with threads requiring variable pitch.

In forthcoming research, currently under preparation, the aim is to broaden the scope of applicability by incorporating cases with variable radius. This extension will provide CNC operators with even greater versatility in machining processes. Simultaneously, there is a plan to delve deeper into the interpretation of helical equations. The goal is to enable the selection of different pitches while maintaining consistent numbers of turns and lengths, thereby offering a nuanced approach to helical geometry.

Funding Open access funding provided by the Cyprus Libraries Consortium (CLC). No funding was received to assist with the preparation of this manuscript.

Declarations

Competing interests The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Kumar K, Ranjan C, Davim J (2020) "Canned cycle. In: CNC programming for machining. Materials forming, machining and tribology". Springer, Cham. https://doi.org/10.1007/978-3-030-41279-1_9

2. Martinov GM, Martinova LI, Fokin NN (2021) Development of toolkit for formalizing the programming of canned cycles on CNC machine tools. MATEC Web Conf 346:03098. https://doi.org/10.1051/matecconf/202134603098
3. Russell L (2023) CNC programming: tutorial examples G & M codes. Liam Russell
4. Araujo AC, Silveira JL (2006) Thrilling: An application of thread milling process. IV Congresso Nacional de Engenharia Mecânica, Recife, Brazil. ⟨hal-03212330⟩. Available at: https://hal.science/hal-03212330/document
5. Cong K, Ma D et al (2023) "Design and Analysis of passive variable pitch propeller for VTOL UAVs" AerospSci Technol, Elsevier 132. https://doi.org/10.1016/j.ast.2022.108063
6. Gustafson PA, Veenstra JM, Bearden CR, Jastifer JR (2019) "The effect of pitch variation and diameter variation on screw pullout. 2(3):258–263. https://doi.org/10.1177/1938640018789999
7. Austin J, Roebke BS, Logan J et al (2018) Fracture gap reduction with variable-pitch headless screws. J Hand Surg 43(4):385.e1-385.e8. https://doi.org/10.1016/j.jhsa.2017.10.018
8. Tao R, Qingyou L, Yonghua C et al (2016) Variable pitch helical drive in-pipe robot. Int J Rob Autom 31(3):263–271. https://doi.org/10.2316/Journal.206.2016.3.206-4774
9. HEIDENHAIN iTNC 530 (2008) CNC control user manual. Available at: https://content.heidenhain.de/doku/tnc_guide/pdf_files/iTNC530/34049x-05/zyklen/670_388-20.pdf
10. HAAS Automation, Inc (2017) Mill operator's manual. Available at: https://www.haascnc.com/content/dam/haascnc/en/service/manual/operator/english---mill-ngc---operator%27s-manual---2017.pdf
11. FANUC Series oi – MC (2014) Operator's manual. Available at: https://edisciplinas.usp.br/pluginfile.php/7532034/mod_resource/content/1/Manual%20Programa%C3%A7%C3%A3o%20CNC%20Fanuc%20Oi-MC%20Fresamento.pdf
12. Abbas AT, Megahed SM (2005) A general algorithm for drilling holes lying in a matrix. Robot Comput- Integr Manuf 21(3):235–239. https://doi.org/10.1016/j.rcim.2004.08.001
13. Omirou S, Fyrillas M (2015) A general G-code algorithm for deep hole drilling. J Manuf Sci Prod 15(2):225–237. https://doi.org/10.1515/jmsp-2014-0028
14. Omirou S (2016) A CNC parametric programming method for manufacturing of axisymmetric mould cavities. J Manuf Sci Prod 16(3):173–181. https://doi.org/10.1515/jmsp-2016-0017
15. Abbas M (2015) "Enhancement of the capabilities of CNC machines via the addition of a new counter boring cycle with a milling cutter", Mech Eng Res 5(2). https://doi.org/10.5539/mer.v5n2p45
16. Rodriguez-Alabanda O, Romero PE, Guerrero-Vaca G (2019) Application of Custom Macro B high level CNC programming language in a five-axis milling machine for drilling holes distributed in axi-symmetric working planes. Procedia Manuf 41:976–983. https://doi.org/10.1016/j.promfg.2019.10.023. (**ISSN 2351-9789**)
17. Lynch M (1997) "Parametric programming for computer numerical control machine tools and touch probes", SocManuf Eng
18. Wolfram Mathworld (2023) https://mathworld.wolfram.com/Helix.html

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.