



# **BRNO UNIVERSITY OF TECHNOLOGY**

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

## **FACULTY OF MECHANICAL ENGINEERING**

FAKULTA STROJNÍHO INŽENÝRSTVÍ

## **INSTITUTE OF MANUFACTURING TECHNOLOGY**

ÚSTAV STROJÍRENSKÉ TECHNOLOGIE

## **SPLINE INTERPOLATION IN CNC MILLING**

TECHNOLOGIE CNC FRÉZOVÁNÍ POMOCÍ SPLAJNOVÝCH INTERPOLACÍ

### **BACHELOR'S THESIS**

BAKALÁŘSKÁ PRÁCE

### **AUTHOR**

AUTOR PRÁCE

**Veronika Soukupová**

### **SUPERVISOR**

VEDOUCÍ PRÁCE

**Ing. Petra Ohnišťová**

**BRNO 2018**

# Bachelor's Thesis Assignment

Institut: Institute of Manufacturing Technology  
Student: **Veronika Soukupová**  
Degree program: Engineering  
Branch: Fundamentals of Mechanical Engineering  
Supervisor: **Ing. Petra Ohnišřová**  
Academic year: 2017/18

As provided for by the Act No. 111/98 Coll. on higher education institutions and the BUT Study and Examination Regulations, the director of the Institute hereby assigns the following topic of Bachelor's Thesis:

## Spline interpolation in CNC milling

### Brief description:

Description of spline interpolations for CNC milling, basic mathematical definition of spline interpolation, generation of a CNC program and its experimental verification.

### Bachelor's Thesis goals:

1. Theoretical analysis
2. Design of a CNC program using spline interpolation
3. Experimental verification of the CNC program with spline application

### Recommended bibliography:

LANGERON, J. M. et al. In A new format for 5-axis tool path computation, using Bspline curves. Computer-Aided Desing. 2004, 36(12), p. 1219-1229.

VAVRUŠKA, P. Machine tool control systems and interpolations of spline type. Engineering MECHANICS. 2012, 19(4), p. 219-229.

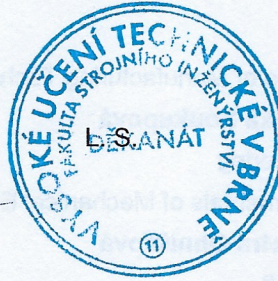
FAVROLLES, Jean-Pierre. Les surfaces complexes: pratique et applications. Paris: Hermes, 1998. ISBN 28-660-1675-0.

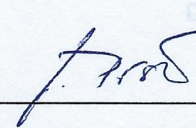
MÜLLER, M., ERDÖS, G., XIROUCHAKIS, P. High accuracy spline interpolation for 5-axis machining. Computer-Aided design. 2004, 36(13), pp. 1379-1393.

WANG, Y. et al. Integration of a 5-axis Spline Interpolation Controller in an Open CNC System. Chinese Journal of Aeronautics. 2009, 22(2), pp. 218-224.

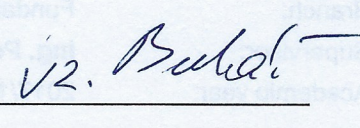
Students are required to submit the thesis within the deadlines stated in the schedule of the academic year 2017/18.

In Brno, 30. 10. 2017



  
\_\_\_\_\_

prof. Ing. Miroslav Píška, CSc.  
Director of the Institute

  
\_\_\_\_\_

doc. Ing. Jaroslav Katolický, Ph.D.  
FME dean

## **ABSTRAKT**

Tato práce se zaměřuje na využití různých typů splinové interpolace pro obrábění na číslicově řízených strojích. Cílem této práce je teoreticky popsat využití splinových interpolací při CNC frézování a následně stanovit a zhodnotit limity využití různých typů splinových interpolací na základně daného objemu vstupních dat. Součástí práce je vytvoření programu pro generaci různého počtu uzlových bodů na analyticky známé křivce. Následuje vložení uzlových bodů do vytvořeného CNC programu pětiosého obráběcího centra a poté jeho verifikace. Pro stejné uzlové body jsou aplikovány různé druhy interpolace pro následné porovnání metod a stanovení minimálního počtu bodů na délku křivky pro dosažené vyhovující přesnosti při použití dané interpolační metody.

### **Klíčová slova**

spline, interpolace, CNC, NC, frézování, CAD, A-spline, B-spline, C-spline, NURBS

## **ABSTRACT**

This study aims to describe application of spline interpolations for computer numerical control machines. The aim of this work is also to evaluate limits of application of different spline interpolation methods and to determine minimum input data for different methods. The thesis includes creation of a program to generate different number of knots on an analytically known curve. Then we input the knots in a NC program for five axes CNC milling machine and verification of the program. Knots are then interpolated by different splines to compare the methods to each other and determine minimal number of knots per curve length to obtain sufficient precision using the chosen method.

### **Key words**

spline, interpolation, CNC, NC, milling, CAD, A-spline, B-spline, C-spline, NURBS

## ROZŠÍŘENÝ ABSTRAKT

Lineární interpolace je často využívaná metoda CNC obrábění a lze s ní dosáhnout velmi přesných tolerancí, pokud je definováno dostatečné množství vstupních dat (řídících bodů). Nicméně pro mnoho aplikací je vhodnější lineární interpolaci nahradit některým typem splinové interpolace. Splinová interpolace je v podstatě proložení uzlových bodů hladkými křivkami (polynomy druhého nebo třetího řádu), díky čemuž je kinematika pohybu nástroje hladší bez nadměrného zrychlování a zpomalování posuvu nástroje, než jak je tomu při využití lineární interpolace. Při splinových interpolacích jsou přechody mezi jednotlivými uzlovými body hladké a nedochází k tvorbě fazetek jako v případě lineární interpolace. Menší objem vstupních dat pro splinové interpolace má dále za následek výrazné zmenšení velikosti NC programů, a tudíž menší nároky na paměť obráběcího stroje. [1] [2].

Splinové interpolace mají velký potenciál uplatnění například v reverzním inženýrství, kdy je součást vyrobena na základě omezeného počtu uzlových bodů definujících geometrii reálné součástky, bez nutnosti mnohdy zdlouhavého skenování, vytváření a optimalizování CAD modelu. Splinové interpolace se dají využít také při CNC obrábění součástí, jejichž geometrie je definována pouze za pomoci analyticky známých křivek. V případech, kdy jsou komplexní tvary obráběných součástí definovány pouze omezeným počtem řídících bodů je nutné tyto body aproximovat vhodnou funkcí tak, aby byla zajištěna nejvyšší přesnost obrábění [1] [3].

Většina dostupných řídících systému jako například Heidenhain, Fanuc nebo Sinumerik umožňují využití základních interpolačních metod. Mezi nejzákladnější interpolační metody patří A-spline, B-spline a C-spline. A-spline je označení pro Akimův spline, B-spline značí Beziérův spline a C-spline je zkratka pro kubický spline. Pro B-spline je charakteristické, že neprochází uzlovými body kromě prvního a posledního uzlu. Většinou je B-spline aplikován ve formě NURBS (Non-uniform rational basis spline) u kterého má každý uzel navíc parametr váhy. Čím větší váha bodu, tím více je splinová křiva přitahována k danému bodu. Nastavením velké váhy uzlů lze zmenšit chybu způsobenou tím, že B-spline uzlovými body přímo neprochází. Pro C-spline je typické, že má ze studovaných tří typů splinů největší sklon oscilovat. Pro A-spline je zase typický lokální aspekt, kdy je křivka v určité oblasti ovlivněna pouze pěti body najednou. Tím pádem A-spline není příliš deformován skokovými změnami a je dobře aplikovatelný pro schodové funkce. Všechny tři typy, jejich definice a vlastnosti byly detailně popsány v teoretické části této práce [4] [5].

Využití splinových interpolací, kdy je geometrie součástky popsána za pomoci analyticky známe křivky, bylo analyzováno v experimentální části této práce. Nejprve byla definována analyticky známá křivka obráběné součásti, na níž bylo generováno různé množství uzlových bodů, které sloužily jako řídící body pro jednotlivé druhy splinových interpolací (A-spline, B-spline a C-spline) a pro lineární interpolaci. Cílem práce bylo stanovit vliv počtu a vzájemné vzdálenosti řídících bodů na přesnost a vhodnost jednotlivých splinových interpolací. Generování řídících bodů na analyticky známé křivce bylo realizováno dvěma různými způsoby za využití matematického softwaru MATLAB 2017b. Prvním způsobem bylo generování bodů s polární distribucí, kdy byl měněn úhel v polárním systému mezi uzly. Byly vytvořeny 3 variace pro úhel mezi každými dvěma sousedními body: 1°, 5° a 15°. Druhým způsobem bylo generování bodů s ekvidistantní

distribucí, kdy byla měněna délka oblouku mezi uzlovými body: 0,38148 mm, 0,76943 mm a 2,38929 mm.

Hlavní program splinových interpolací byl vytvořen v řídicím systému Sinumerik 840i. Řídicí body vygenerované v programovací platformě MATLAB 2017b a MATLAB 2018a sloužily jako NC podprogramy pro splinové interpolace a lineární interpolaci. Experimentální ověření programů bylo realizováno na pětiosém frézovacím centru TAJMAC-ZPS MCV 1210 pro 24 různých variant obráběných kontur.

Pomocí programu MATLAB bylo dále vygenerováno 11 kontrolních bodů v druhém kvadrantu příslušného kartézského souřadného systému. Souřadnice teoretických kontrolních bodů byly porovnány s pozicemi obroběných kontur, které byly naměřeny pomocí dílenského měřicího mikroskopu MarVision MM 420. Byly porovnány rozdíly mezi souřadnicemi jednotlivých bodů a také byla spočítána průměrná absolutní chyba každé metody. Porovnáním průměrných absolutních chyb všech použitých interpolačních metod pro stejný typ distribuce uzlů jsme získali přehled použitelnosti interpolačních metod pro použité variace distribuce uzlů na kontuře. Výsledky měření a zhodnocení použitelnosti dané interpolační metody pro danu distribuci uzlů jsou zaznamenány v tabulkách 1 a 2

**Tabulka 1** Použitelnost splinových interpolací pro ekvidistantní distribuci uzlů na kontuře.

Interpoační metoda	Distribuce uzlů	Počet uzlů	Vzdálenost mezi uzly (mm)	Průměrná odchylka (mm)	Požitelnost
A-spline	Ekvidistantní	357	0,38148	0,057	ANO
B-spline	Ekvidistantní	357	0,38148	0,047	ANO
C-spline	Ekvidistantní	357	0,38148	0,059	ANO
Linear	Ekvidistantní	357	0,38148	0,033	ANO
A-spline	Ekvidistantní	177	0,76943	0,025	ANO
B-spline	Ekvidistantní	177	0,76943	0,055	ANO
C-spline	Ekvidistantní	177	0,76943	0,046	ANO
Linear	Ekvidistantní	177	0,76943	0,040	ANO
A-spline	Ekvidistantní	57	2,38929	0,050	ANO
B-spline	Ekvidistantní	57	2,38929	0,330	NE
C-spline	Ekvidistantní	57	2,38929	0,070	ANO
Linear	Ekvidistantní	57	2,38929	0,114	NE

**Tabulka 2 Použitelnost splinových interpolací pro polární distribuci uzlů na kontuře.**

Interpoační metoda	Distribuce uzlů	Počet uzlů	Úhel mezi uzly	Průměrná odchylka (mm)	Použitelnost
A-spline	Polární	360	1	0,050	ANO
B-spline	Polární	360	1	0,099	NE
C-spline	Polární	360	1	0,049	ANO
Linear	Polární	360	1	0,054	ANO
A-spline	Polární	72	5	0,041	ANO
B-spline	Polární	72	5	0,234	NE
C-spline	Polární	72	5	0,035	ANO
Linear	Polární	72	5	0,088	NE
A-spline	Polární	24	15	0,676	NE
B-spline	Polární	24	15	1,241	NE
C-spline	Polární	24	15	0,382	NE
Linear	Polární	24	15	0,665	NE

Pro oba typy distribuce uzlových bodů se C-spline interpolace prokázala jako nepřesnější metoda, následována A-spline interpolací. B-spline vykazoval velké nepřesnosti a křivku deformoval nejvíce.

Limitní hodnoty (ze zkoumaných hodnot) pro použití splinových interpolačních metod pro obráběnou konturu, kdy je daná metoda aplikovatelná jsou:

- pro A-spline: ekvidistantní distribuce s 2,38929 mm mezi jednotlivými uzly,
- pro B-spline: ekvidistantní distribuce s 0,76943 mm mezi jednotlivými uzly,
- pro C-spline: ekvidistantní distribuce s 2,38929 mm mezi jednotlivými uzly,

pro lineární interpolaci: ekvidistantní distribuce s 0,76943 mm mezi jednotlivými uzly.

Vliv zvolené interpolační metody na strojní čas byl také prozkoumán

Výhody využití splinové interpolace oproti lineární interpolaci jsou následující:

- hladká trajektorie nástroje a menší drsnost povrchu,
- snížení strojního času
- pro stejnou přesnost spline interpolacím stačí menší objem vstupních dat a tedy menší velikost NC programů.

**BIBLIGRAPHIC CITATION**

SOUKUPOVÁ, V. *Technologie CNC frézování pomocí splajnových interpolací*. Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství, 2018. 129 s., Vedoucí bakalářské práce Ing. Petra Ohnišťová.



**AFFIRMATION**

I declare that this bachelor thesis is result of my own work, led by my supervisor, and all used sources are duly listed in bibliography. I proclaim that all presented information is true and valid to the best of my knowledge.

---

Date

---

Veronika Soukupová

## PODĚKOVÁNÍ

Tímto bych ráda poděkovala vedoucí své bakalářské práce Ing. Petře Ohnišťové za veškeré rady a připomínky během vypracování této studie, dále prof. Ing. Miroslavovi Píškovi, CSc. za rady a podporu během studia v Cluny. Děkuji také panu Jiřímu Čechovi za ochotu a trpělivost při spolupráci na experimentální části práce.

Za morální podporu během psaní této bakalářské práce děkuji Kateřině Wieckové, Hanně Havlíkové, Katrin Bučkové, Tomášovi Raškovi a Léonardovi Suslianovi.

Speciální poděkování patří mé rodině za veškerou podporu během celého mého studia.

## ACKNOWLEDGMENT

I would like to thank the supervisor of my bachelor's thesis Petra Ohnišťová for all advice and remarks during the creation of this study. I would also express my gratitude towards prof. Ing. Miroslav Píška, CSc for all advice and support during my studies in Cluny. I also thank Mr. Jiří Čech for his willingness and patience during the cooperation on the experimental part of this thesis.

For moral support provided during writing this thesis I would like to express my gratitude to Kateřina Wiecková, Hanna Havlíková, Katrin Bučková, Tomáš Raška and Léonard Suslian.

A special acknowledgement belongs to my family for overall support during all of my studies.

**CONTENTS**

ABSTRAKT .....	4
INTRODUCTION .....	14
1 THEORETICAL ANALYSIS .....	15
1.1 CNC MILLING.....	15
1.1.1 Basic principles of CNC .....	15
1.1.2 Milling .....	15
1.1.3 CNC milling.....	17
1.1.4 Coordinate systems in CNC machine .....	17
1.1.5 Trajectory programming .....	19
1.2 Types of interpolation .....	20
1.2.1 Linear .....	20
1.2.2 Circular .....	23
1.2.3 Spline interpolation.....	23
1.2.3.1 A-spline .....	24
1.2.3.2 B-spline.....	26
1.2.3.3 C-spline.....	29
1.3 NC programs .....	31
1.3.1 Types of programming.....	31
1.3.1.1 Absolute programming.....	31
1.3.1.2 Incremental programming .....	31
1.3.2 Structure of the program .....	32
1.3.2.1 Letter addresses .....	32
1.3.2.2 G functions .....	33
1.3.2.3 M functions.....	34
1.4 Spline interpolation in different control systems .....	34
1.4.1 Sinumerik.....	34
1.4.1.1 Programming B-splines in Sinumerik .....	35
1.4.1.2 Programming A-spline and C-spline in Sinumerik .....	36
1.4.2 Heidenhain .....	38
1.4.3 Fanuc.....	39
1.5 Comparison of interpolation methods for CNC .....	39

2	Design of a CNC program using spline interpolation.....	41
2.1	Creation of the contour.....	41
2.1.1	Curve in polar coordinate system .....	42
2.1.2	Choosing an equation of the contour .....	43
2.1.3	Arc length .....	54
2.1.4	Minimal radius of curvature .....	54
2.2	Generating knots .....	55
2.2.1	Polar distribution.....	56
2.2.2	Equidistant distribution.....	59
2.2.3	2D and 3D contouring.....	66
2.2.3.1	2D contouring.....	66
2.2.3.2	3D contouring.....	67
2.3	Creation of programs.....	70
2.3.1	Methodology of creation of subprograms.....	71
2.3.2	Programs for 2D contouring .....	71
2.3.2.1	Rough specimen for 2D contouring .....	71
2.3.2.2	Finishing for 2D contouring .....	73
2.3.3	Programs for 3D contouring .....	74
2.3.3.1	Rough specimen for 3D solution.....	74
2.3.3.2	Finishing for 3D contouring .....	76
3	Experimental verification of the CNC program with spline application .....	78
3.1	CNC machine .....	78
3.2	Workpiece .....	79
3.2.1	Workpiece for 2D contouring.....	79
3.2.2	Workpiece for 3D contouring.....	79
3.3	Selection of the tool .....	80
3.4	Selection of cutting conditions.....	80
3.5	Procedure of the execution.....	80
3.6	Program simulation .....	81
3.7	Experimental verification of spline interpolation on 2D contouring .....	81
3.7.1	Clamping and coordinate system.....	81

3.7.2	Rough specimen execution of 2D contour.....	83
3.7.3	Finishing execution of 2D contour .....	85
3.8	Problems with 2D contours .....	85
3.9	Using 3D contouring as a solution .....	88
3.10	Methodology of evaluating the surface positions .....	89
3.11	Observations.....	93
3.12	Interpretation of measurements of surface position .....	101
3.12.1	Error in individual control points.....	101
3.12.2	Absolute mean error of methods.....	103
3.12.3	Accuracy of spline interpolations for polar distribution.....	104
3.12.3.1	Error of spline interpolations for polar distribution in individual control points	104
3.12.3.2	Mean absolute error of spline interpolations for polar distribution.....	107
3.12.4	Accuracy of spline interpolations for equidistant distribution.....	110
3.12.4.1	Error of spline interpolations for equidistant distribution in individual control points .....	110
3.12.4.2	Mean absolute error of spline interpolations for equidistant distribution	114
3.12.5	Imperfection of used evaluation method .....	117
3.13	Influence of spline interpolation method on machining time .....	118
3.14	Discussion .....	120
	Conclusions.....	123
	List of symbols and abbreviations .....	127
	List of appendices .....	129

## INTRODUCTION

Linear interpolation is a widely used method for CNC milling and can result in very accurate results when provided sufficient volume of input data. Nevertheless for many applications spline interpolations has potential to replace linear interpolation as they bring multiple advantages such as more perfect kinetics and lesser input data needed to obtain precise machined curve. Lesser input data mean smaller NC programs. Spline interpolation represent great potential to be employed in reverse engineering where a part is reproduced based on a pre-existing part possibly avoiding usage of CAD/CAM systems to create a numerical model firstly which can be a lengthy process. Other possible application is to machine analytically known curves while also skipping the CAD/CAM systems. [1, 2].

The latter application was tested as the experimental part of this study. A theoretically known contour was created and different numbers of points (also called knots in terms of interpolation) were generated as input data. Creation and execution of programs containing the generated knots was set as a goal. Multiple varieties of input data were designed to compare accuracy of three types of spline interpolation methods (A-spline, B-spline and C-spline) for CNC milling. Number of knots was varied as well as the type of distribution of the knots along the contour. How these parameters influenced the applicability of chosen spline interpolation method was evaluated and limits for each method were established.

# 1 THEORETICAL ANALYSIS

## 1.1 CNC MILLING

### 1.1.1 Basic principles of CNC

Computer Numerical Control abbreviated CNC represents the automation of machine tools and it aims to partially or entirely substitute manually controlled machining. In general, a CNC machine consists of following parts: machine tool itself and a control system. The machine tool structure consists of base, column, spindle, worktable, milling head and a tool [1].

Main objective of a CNC machine is to guide the tool path respectively to the machine's coordinate system to machine the workpiece accordingly to the requirements [1].

Structure of CNC machines usually contains detectors to control different stages of the machining process. We differentiate three groups of detectors integrated in a CNC system [1]:

- empiric detectors,
- presence detectors,
- real-time detectors.

A CNC machine performs a sequence of commands based on the directives in the program. CNC machines can operate with minimal intervention of the operator or even unattended. The operator is needed for initial adjustments and fixation of the workpiece, therefore the skill level required of the operator is lowered compared to manual machining. Multiple machining operations can be performed in one workpiece clamping. Another major advantage of CNC is its high degree of accuracy ensuring repeatable and consistent results. Minimal intervention of the operator minimizes human error and therefore contributes to consistent results [3].

CNC machines are usually equipped with an automatic tool changer (ATC) that allows rapid change of tools which significantly decreases the machining time for operations that require tool change [3].

Replacing manual machining with CNC brings general productivity increase and better precision [3].

### 1.1.2 Milling

Milling can be defined as a machining process that removes material from the surface of the workpiece by a milling cutter which performs a rotary movement while either the workpiece or the tool is advancing. Thus the main cutting movement is performed by the tool, and secondary movement can be performed either by the workpiece or by the tool. The cutting movement is discontinuous and milling cutters usually have multiple blades [4]. Diagram of general milling is displayed on Figure 1.1.2.1.

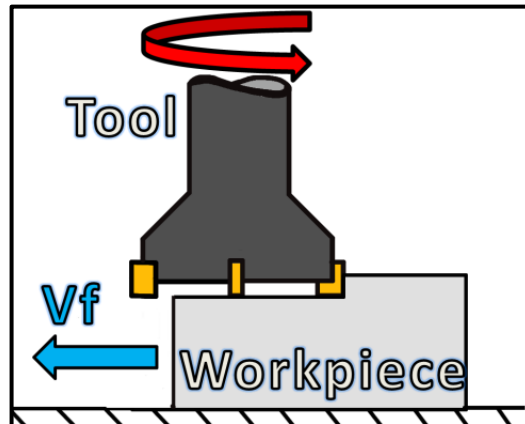


Figure 1.1.2.1 Milling diagram.

In case of so called conventional milling the workpiece is fed against the milling cutter [4]. The chip cross section progress from minimum to maximum as displayed in Figure 1.1.2.1.

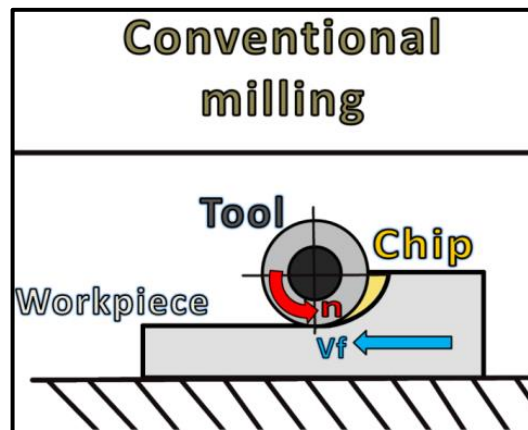


Figure 1.1.2.2 Conventional milling.

In case of climb milling the workpiece fed with the milling cutter. The chip cross section progress from maximum to minimum as is displayed in Figure 1.1.2.3 [4].

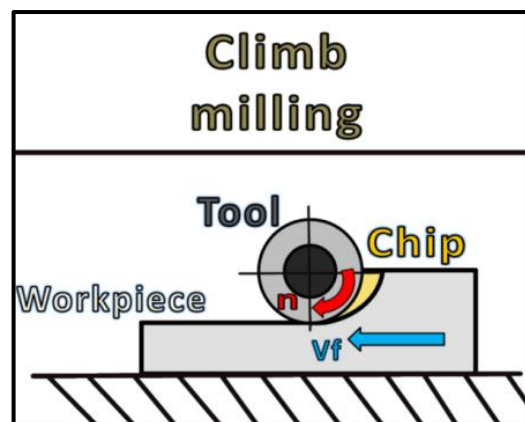


Figure 1.1.2.3 Climb milling.



### 1.1.3 CNC milling

CNC lathes and CNC milling machines can be differentiated. CNC lathes are used to machine rotational workpieces while CNC mills are in addition able to machine non-rotational pieces. CNC milling machines with multiple axes are usually identifiable as milling machines. CNC milling machines are widely used thanks to their versatility to machine both rotational non-rotational surfaces [4].

Milling Machines can be classified by different criteria [3]:

- by the number of axes: the number of axes varies from two to five or even more,
- by the orientation of its linear axes: linear axes can be vertically or horizontally oriented, example of a vertical mill is displayed on Figure 1.2.3.1,
- by the presence or absence of an automatic tool changer (ATC): vast majority of CNC machines are equipped with an ATC. There are various types of ATC system.

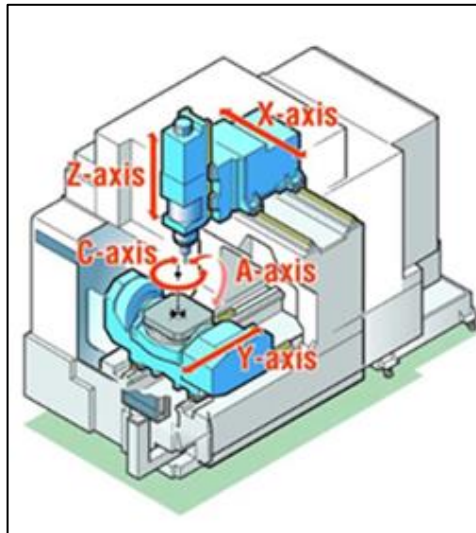


Figure 1.1.3.1 Vertical CNC mill Mori Seiki GV 5035AX [4].

### 1.1.4 Coordinate systems in CNC machine

CNC machines guide the reciprocal tool-workpiece movement in a coordinate system. CNC machines most commonly use right handed Cartesian coordinate system of pair-wise orthogonal axes X, Y, Z which are parallel to leading surfaces of the machine [5]. Different axis must correspond to the norm NF ISO 841:2004-09 [1].

For milling: Z axis usually corresponds to the axis of the tool. The term horizontal mill refers to a Z axis mill. X axis should also be horizontal if possible [1]. Five axes CNC machines have also two rotational axes, either A, B or B, C or A, C [3].

Representation of a right handed Cartesian coordinate system with three correspondent rotational axis is displayed on Figure 1.1.4.1.

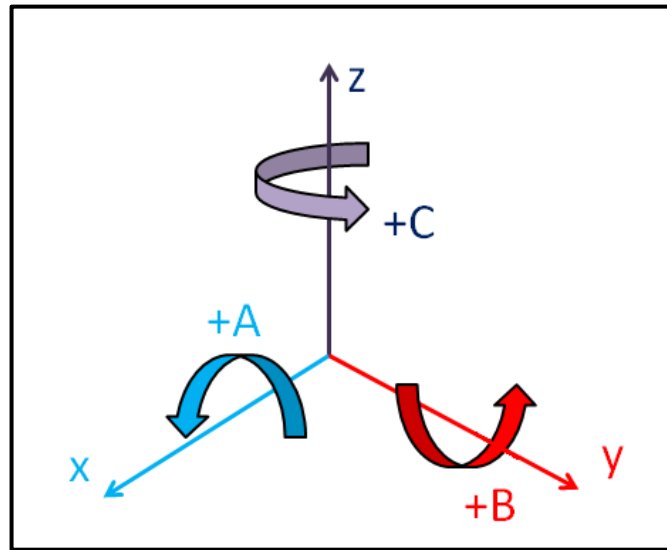


Figure 1.1.4.1 Axes X, Y, Z, A, B, C of a coordinate system.

Rotation can be achieved in multiple configurations. Either the workpiece or the tool can rotate around different axes. Reference points are placed in the coordinate system to determine tool-workpiece position [5]:

- M: machine reference point, origin of the machine coordinate system, fixed by the machine manufacturer, cannot be modified by the user,
- R: reference point – a point fixed by the machine manufacturer use for precise measurements and to set other in the course of the program,

W: workpiece zero point – origin of the workpiece's coordinate system in which all instruction in the program are referred to. Workpiece zero point is determinate by the creator of the program and can be re-defined at any point of the program,

- T: tool mount reference point – the machine operator must enter the distance between point T and the tip of the tool, every tool in the ATC needs to have this distance measured in order to be usable,
- P: tool setup point – beginning of the toolpath, after all tasks in the NC program are carried out, the tool returns to this point.

Some of the reference points are displayed on Figure 1.1.4.2.

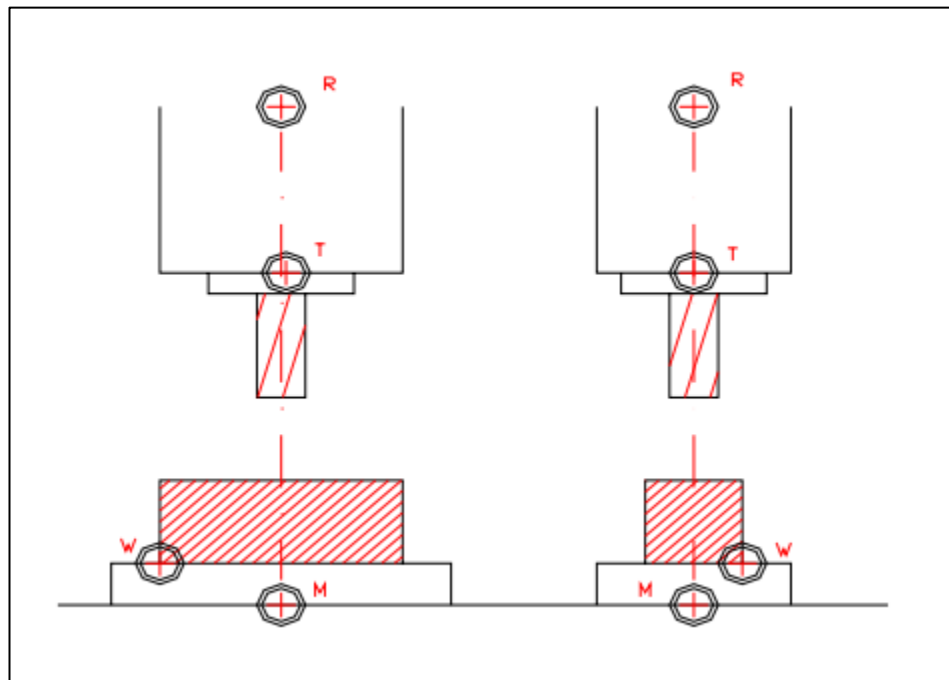


Figure 1.1.4.2 Reference points [5].

### 1.1.5 Trajectory programming

CNC machining must meet functional specifications of the workpiece set by its designer [6].

CNC machines are operated by a numerical control (NC) program. 3D CAD model needs to be mathematically described and transformed into NC data readable by the machine. Creation of NC data consists of four stages. In the first stage the designer creates a CAD model of the workpiece. In the second stage the tool trajectories are calculated in a CAM system in form of CL data. In the third stage CL data are treated by the postprocessor. Postprocessor can be either a part of CAM system or independent. Nevertheless, each postprocessor is designed for a specific pair of CAM system and CNC control system which may cause problems with compatibility. In the fourth stage the NC program is transferred to the CNC machine where it's executed as movements [6, 7, 8]. The process is visualized on Figure 1.1.5.1.

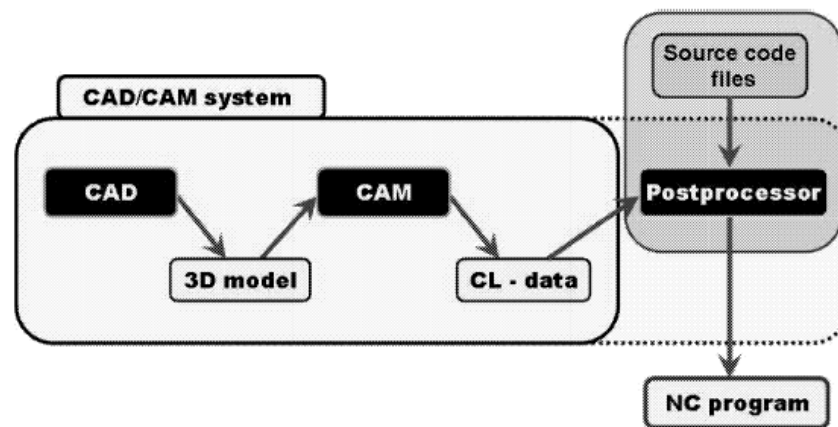


Figure 1.1.5.1 Numerical control program creation process [7].

## 1.2 Types of interpolation

In case of more complex machining, toolpaths are generally freeform curves. Any curved contour of the workpiece can be approximated by a limited number of data points also called knots. These can be connected by different types of curves [9].

Interpolation is a mathematical method that consists of defining a function that passes through  $n + 1$  number of ordered data points:  $P_i(x_i, y_i)$ ,  $i = 0, 1, \dots, n$  while  $x_0 < \bar{x} < x_n$  [3]. If we search to determine  $f(x)$  in  $\bar{x}$ , while  $\bar{x} \neq x_i, i = 0, 1, \dots, n$  it's called an interpolation problem if  $x_0 < \bar{x} < x_n$ . On the contrary if  $\bar{x} < x_0$ , it's called an extrapolation problem [10]. There are several methods of interpolation that vary in accuracy, smoothness and number of data points needed. Following section describes several interpolation methods which can be used to guide toolpath in CNC milling.

### 1.2.1 Linear

Linear interpolation or spline of the first order is the simplest of all interpolations. In case of linear interpolation, the ordered data points are connected by straight lines [10]. The toolpath curves are approximated by multiple lines in a process generally called discretization [11].

Linear interpolation is commonly used because it's easy to establish [6].

When using linear interpolation, the tool is moving point to point following a straight line. The error highly depends on sampling, thus on number of points. The error decreases with increasing number of points. When the tool is passing through the control points the error between the desired curve and the machined profile is null in said point. Factors influencing the value of geometrical error in all other parts are tool blade geometry and feed per dent [6].

The continuous function connecting  $n + 1$  knots  $P_0, P_1, \dots, P_n$  is composed of straight linear segments  $P_i, P_{i+1}$ ,  $i = 0, 1, \dots, n - 1$ . The linear interpolation function is defined as described in equation (1) [12]:

$$f_1(x) = y_0 \frac{x-x_1}{x_0-x_1} + y_1 \frac{x-x_0}{x_1-x_0} \quad x_0 \leq x \leq x_1$$

$$f_2(x) = y_1 \frac{x-x_2}{x_1-x_2} + y_2 \frac{x-x_1}{x_2-x_1} \quad x_1 \leq x \leq x_2 \quad (1)$$

$$f_n(x) = y_{n-1} \frac{x-x_n}{x_{n-1}-x_n} + y_n \frac{x-x_{n-1}}{x_n-x_{n-1}} \quad x_{n-1} \leq x \leq x_n$$

Linear interpolation is represented on Figure 1.2.1.1

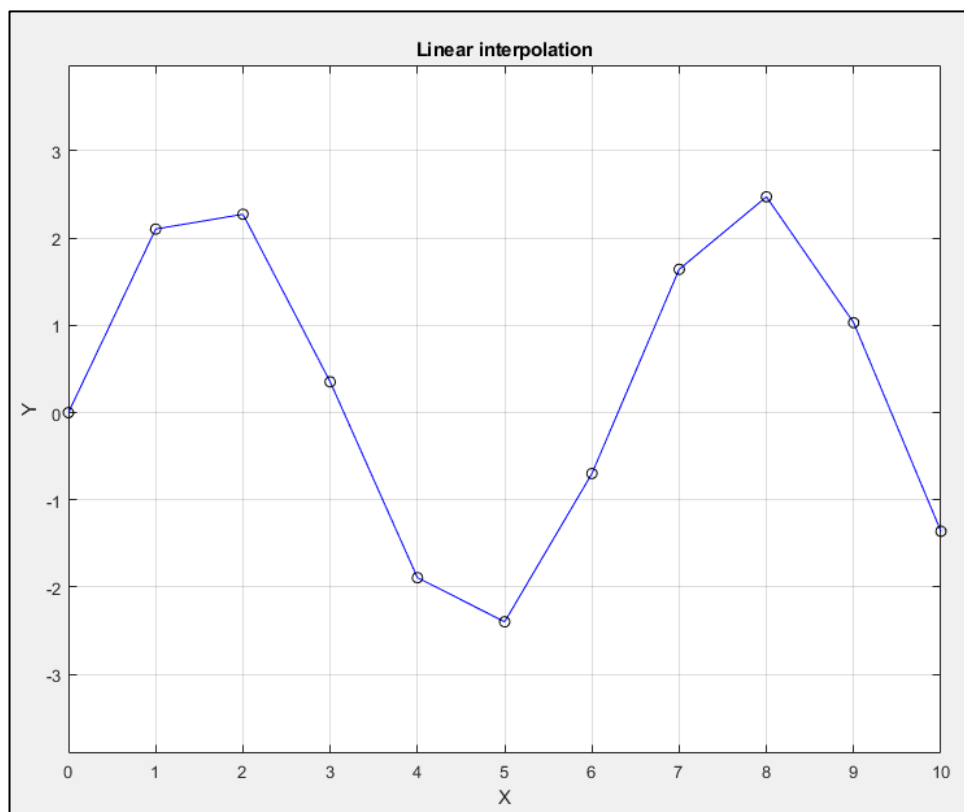


Figure 1.2.1.1 Linear interpolation.

To illustrate the principle of linear interpolation, points situated on a circle with radius equal to  $r = 5$  were generated and connected with segments. On Figure 1.2.1.2, linear interpolation was used to try to obtain interpolation of a circle.

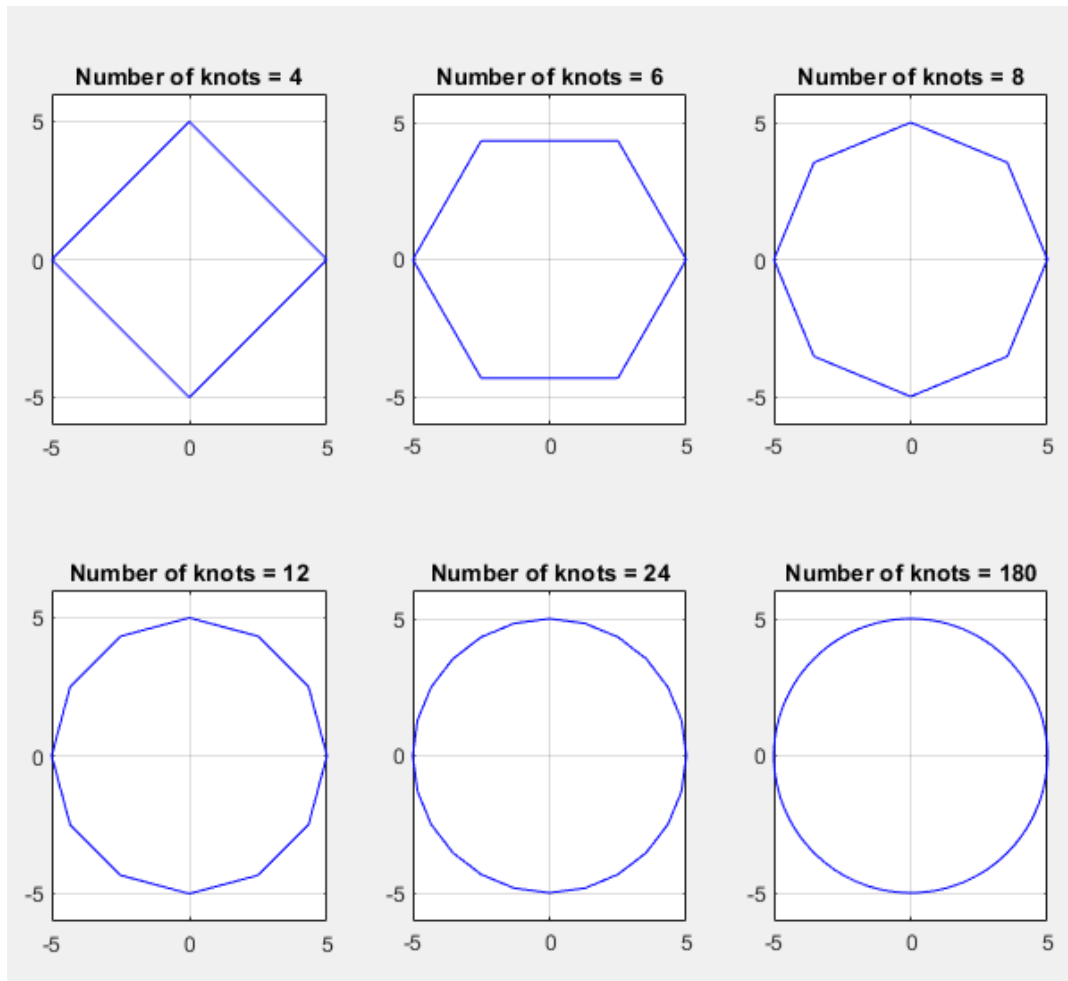


Figure 1.2.1.2 Linear interpolation of knots situated on a circle for different number of knots.

As can be seen on Figure 1.2.1.2, with increasing number of knots per length, the obtained curve gets closer and closer to an actual circle.

Linear interpolation brings multiple disadvantages. The tool stops in every knot then accelerate while moving to the next knot where it stops again. Therefore the velocity is discontinuous in the junctions of every linear segment and high acceleration is needed. As a result we obtain poorer quality surface finish and low accuracy. The fact that the tool is repeatedly accelerating and decelerating with high frequency, results in long machining time. The kinematics of linear interpolation has very low efficiency [11, 13].

To obtain better surface finish, we must shorten the length of segments between two control points, i.e. use finer sampling. This results in greater number of control points which significantly enlarges the size of NC data files that need to be transferred to the controller. The large size of NC files is a major disadvantage of linear interpolation because the memory of the CNC machine can process only a limited amount of data [6].

### 1.2.2 Circular

Circular interpolation is a special type of interpolation that connects two data points by an arc defined by its centre and its radius. Circular interpolation can operate either in a clockwise or counter-clockwise motion [6].

For example in Siemens Sinumerik control systems, there are two functions that call circular interpolation in the program [4]:

- G02: Clockwise circular interpolation,
- G03: Counter-clockwise circular interpolation.

Sinumerik control system enables us to program circular interpolation in three planes. The plane in which the circular interpolation is going to be performed is chosen by one of following functions [4]:

- G17: picks X-Y plane for circular interpolation,
- G18: picks X-Z plane for circular interpolation,
- G19: picks Y-Z plane for circular interpolation.

Some control systems are also able to perform 3D circular interpolation. Machined curve then can be located in any plane given by the programmer, not only in the main planes of the machine's coordinate system [6]. We recognize internal and external circular interpolation. Internal circular interpolation is commonly used to enlarge holes [14].

### 1.2.3 Spline interpolation

Curved parts of the workpiece might not be analytically describable but only approximated by a certain number of control points (knots). Spline defines a curve consisting of parts of polynomials of second or third order. How this curve is dependent on its knots differs for every type of spline. Spline interpolation can be applied to connect digitized points through a smooth curve. The smoothness of the spline is brings significant improvement compared to linear interpolation [9, 15].

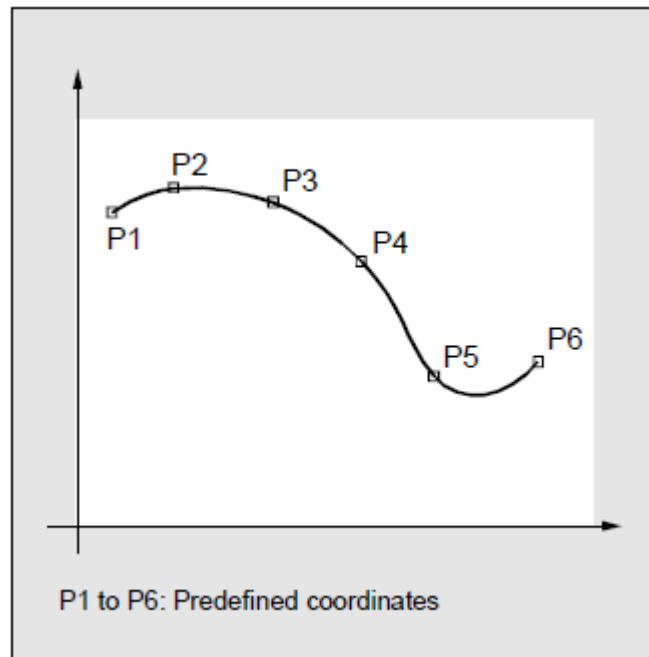


Figure 1.2.3.1 Spline creating a smooth curve connecting 6 predefined knots [15].

### 1.2.3.1 A-spline

Term A-spline stands for Akima spline. It was introduced by Akima in 1977 as a new interpolation method solving problems with undesired wiggles of the curve due to oscillations. Akima proposed to use local aspect as if we tried to draw the curve by hand. That means that the pieces of the curve are influenced only by a small number of neighbour knots [16].

A-spline is a piecewise function composed of third degree polynomials smoothly connected by a geometrical condition. Slope of the curve at every point is determined by only five successive points where the concerned point is the middle with two points at each side [16]

Using only five points at a time eliminates error from extreme differences between neighbour knots, the error will show only locally and not on the whole curve meaning the A-spline rarely oscillates which is a major advantage [16], [13].

Because of the local aspect, changing one control point will affect at most six neighbour points [9].

With five points designated successively 1, 2, 3, 4, 5 we can determine the slope in the central point 3 in multiple ways. The slope of point 3 in A-splines is calculated using equation (2) [16].

$$t_3 = \frac{(|m_4 - m_3| * m_2 + |m_2 - m_1| * m_3)}{(|m_4 - m_3| + |m_2 - m_1|)} \quad (2)$$

Where  $m_1, m_2, m_3, m_4$  stand for the slopes of line segments  $\overline{12}, \overline{23}, \overline{34}, \overline{45}$  [16].



This approach is not applicable in a special case when  $m_1 = m_2 \neq m_3 = m_4$ . In this particular case, the value of  $t$  is by convention equated to  $\frac{1}{2}(m_2 + m_3)$  [16].

Interpolation between two knots with coordinates  $(x_1, y_1)$  and  $(x_2, y_2)$  using the Akima spline consist of connecting them with a polynomial function that passes directly through them and slopes in the two knots are calculated as expressed in (3) [16].

To calculate the third degree polynomial between the two knots following two conditions for every knot are used [16]. First condition ensures that the new polynomial function is going to pass through the knots. In the second condition the slope in the knot is equalled to the first derivative of the new polynomial. By applying these two principles we obtain two pairs of following conditions (3), (4).

$$y = y_1 \quad \text{and} \quad \frac{dy}{dx} = t_1 \quad \text{for } x = x_1 \quad (3)$$

$$y = y_2 \quad \text{and} \quad \frac{dy}{dx} = t_2 \quad \text{for } x = x_2 \quad (4)$$

While these four conditions are enough to describe a unique third degree polynomial, this polynomial can be expressed in other ways [16].

Akima chose following form (5) to describe the third degree polynomial connecting two knots of the Akima spline:

$$y = p_0 + p_1(x - x_1) + p_2(x - x_1)^2 + p_3(x - x_1)^3 \quad (5)$$

Where coefficients are equal to expressions stated in (6) [16].

$$\begin{aligned} p_0 &= y_1 \\ p_1 &= t_1 \\ p_2 &= \left[ \frac{3(y_2 - y_1)}{x_2 - x_1} - 2t_1 - t_2 \right] / (x_2 - x_1) \\ p_3 &= \left[ t_1 + t_2 - \frac{2(y_2 - y_1)}{x_2 - x_1} \right] / (x_2 - x_1)^2 \end{aligned} \quad (6)$$

While calculating the slope at one knot only two preceding knots and two subsequent knots are used. This meant that for the last point of the Akima spline two additional knots must be estimated which leads to potential error in this area [16].

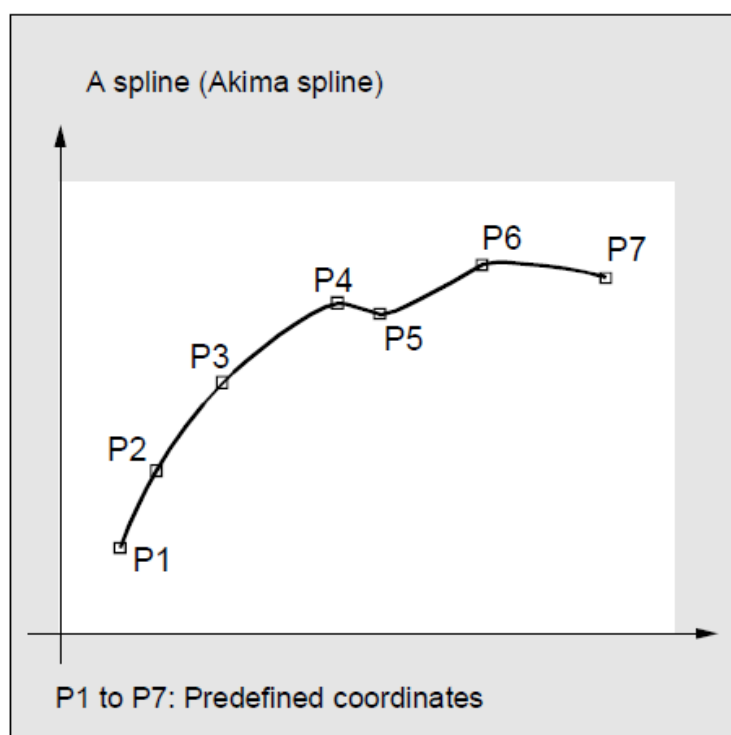


Figure 1.2.3.1.1 Representation of the A-spline interpolating 7 knots [15].

A-spline is especially well applicable for interpolation of curves with great changes of slopes and step (staircase) functions [9].

### 1.2.3.2 B-spline

Term B-spline stands for Bezier spline. B spline consists of pieces of Bezier curves and does not pass through the control points [13].

To understand the nature of B-spline we must first define what Bezier curves are. Parametrical equation of Bezier curve is designated by equation (7) [17].

$$P(u) = \sum_{i=0}^p P_i B_{i,p}(u) \quad \forall u \in [0,1] \quad (7)$$

Expressions  $B_{i,p}(u), \forall u \in [0,1]$ , represent basic (blending) functions and are calculated by (8). It's an ensemble of  $p+1$  Bernstein basis polynomials of degree  $p$ . Expression  $P_i = [P_{xi}, P_{yi}]^T$  designate a control point calculated using (7) [18].

$$B_{i,p}(u) = \frac{p!}{i!(p-i)!} u^i (1-u)^{p-i} \quad 0 \leq i \leq p \quad (8)$$

Example of blending functions is demonstrated in (9) [19].

$$\begin{aligned} \forall u \in [0,1] \\ B_{0,1}(u) &= 1 - u \\ B_{0,2}(u) &= (1 - u)^2 \end{aligned} \quad (9)$$

Functions  $B_{i,n}(t)$  are polynomials of order  $n$  and they form a vector space of polynomials of order inferior or equal to  $n$ .

As can be seen on the Bezier curve passes only through the first control point  $P_0$  and the last control point  $P_n$ . The overall shape of the Bezier curve is determined by other (interior) control points [17].

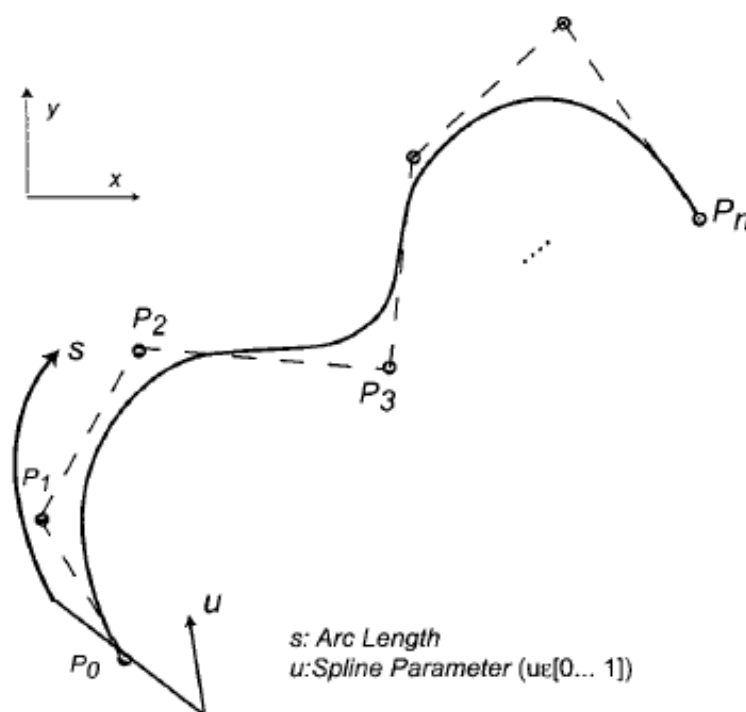


Figure 1.2.3.2 Bezier curve [17].

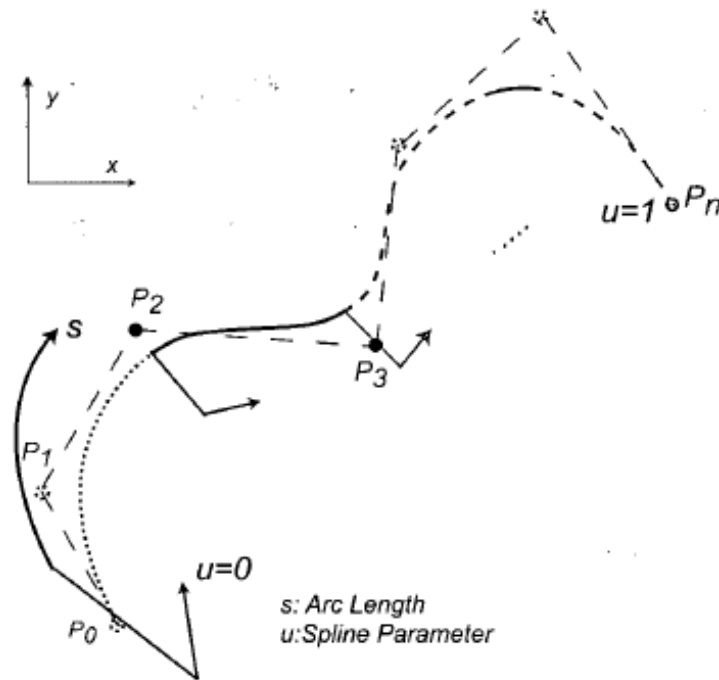


Figure 1.2.3.3 Representation of B-spline [17].

With increasing number of control points the order of the Bezier curve increases as well. To be able to use only lower order curves the solution of B-spline was introduced by Boor for the first time as follows in (10) [17].

$$P(u) = \sum_{i=0}^n N_{i,n}(u)P_i \quad (10)$$

While  $N_{i,n}(u)P_i$  represent new blending functions. B-spline is segmented by a knot vector  $\vec{U}$  into parameter intervals. The curves in these intervals can be modified by nearest knots.

Most commonly used form of B-spline is NURBS (Non-uniform rational basis spline). NURBS is a type of B-spline with the option to add weight to every knot. The greater the weight is the more is the curved “pulled” more towards the knot. Nevertheless even NURBS never passes directly through the points with the exception of the first and the last point. That’s why the curve changes dramatically when we provide lesser knots.

B-splines are well compatible with CAD system. Most CAD systems use B-splines to generate complex surfaces [17].

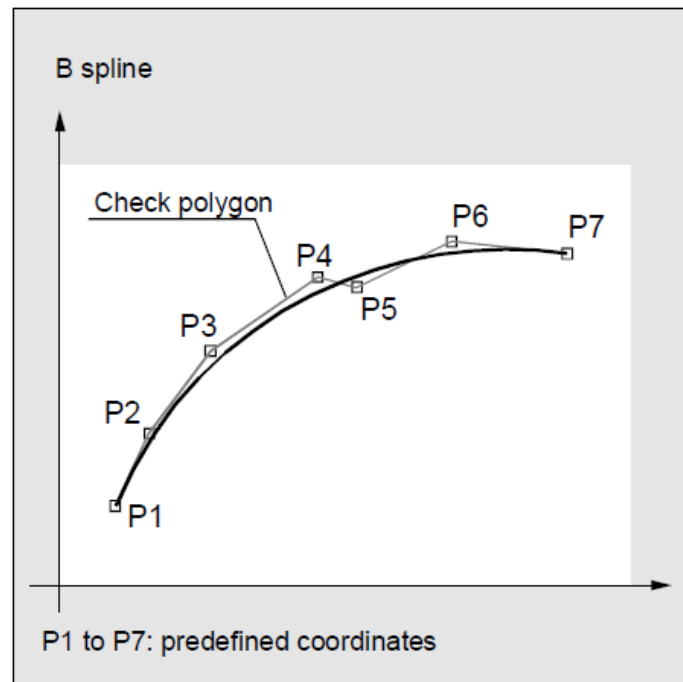


Figure 1.2.3.4 Representation of B-spline interpolating 7 knots [15].

### 1.2.3.3 C-spline

Term C-spline stands for cubic spline. It directly passes through data points and is continuous in each of them. That's why its first and second derivation (curvature) also needs to be continuous in all data points. That is also a reason why it is the most susceptible to be influenced by deviation of (x) out of all studied splines and is likely oscillate in the proximity of extreme values of  $f(x)$  [12], [13].

C-splines are very well applicable if the knots appertain to analytically known functions, for example conic sections or circles [9].

C-spline assesses third degree polynomials for every interval between two knots  $[x_i, x_{i+1}]$  [12].

Third degree polynomials are described as follows in equation (11) [12].

$$f_i = a_i x^3 + b_i x^2 + c_i x + d_i \quad x_{i-1} \leq x \leq x_i \quad i = 1, 2, \dots, n \quad (11)$$

The second derivative in  $x_i$  is designated as  $f_i''$ . First and second derivatives must be continuous. Condition of continuity of the second derivative is defined as (12) [12].

$$f_i''(x_{i-1}) = f_{i-1}'' \quad f_i''(x_i) = f_i'' \quad i = 1, 2, \dots, n \quad (12)$$

Because the second derivative is linear, it is equal to (13):

$$f_i''(x) = f_{i-1}'' \frac{x - x_i}{x_{i-1} - x_i} + f_{i-1}'' \frac{x - x_{i-1}}{x_i - x_{i-1}} \quad (13)$$

Expression  $(x_i - x_{i-1})$  is then replaced by  $h_i$  to simplify the previous expression (13) and we obtain (14) [12]:

$$f_i''(x) = f_{i-1}'' \frac{x_i - x}{h_i} + f_{i-1}'' \frac{x - x_{i-1}}{h_i} \quad (14)$$

By integrating (14) two times and calculating integration constants equation (15) is obtained [12].

$$f_i(x) = f_{i-1}'' \frac{(x_i - x)^3}{6h_i} + f_i'' \frac{(x - x_{i-1})^3}{6h_i} + \left[ \frac{y_{i-1}}{h_i} - f_{i-1}'' \frac{h_i}{6} \right] (x_i - x) + \left[ \frac{y_i}{h_i} - f_i'' \frac{h_i}{6} \right] (x - x_{i-1}) \quad (15)$$

In expression (15)  $f_{i-1}''$  and  $f_i''$  are the only unknown variables. The polynomials  $f_i(x)$  can be determined once we find their second derivatives,  $f_i''$ . To determine the values of  $f_i''$  the condition of continuity of first derivatives (16) is used [12].

$$f_i'(x_i) = f_{i+1}'(x_i) \quad i = 1, 2, \dots, n - 1 \quad (16)$$

By substituting (14) in (15) we obtain (17) [12]:

$$h_i f_{i-1}'' + 2(h_i + h_{i+1}) f_i'' + h_{i+1} f_{i+1}'' = \frac{6}{h_{i+1}} (y_{i+1} - y_i) + \frac{6}{h_i} (y_{i-1} - y_i) \quad (17)$$

$$i = 1, 2, \dots, n$$

While  $n$  being the number of knots, we get  $n-1$  linear equations for  $n+1$  unknown values of  $f_i''$ . Two additional conditions must be added. Two conditions for second derivatives on the boundary of the interpolation interval were set. First condition (18) is for the first polynomial function in the first knot  $x_0$  and a second condition (19) for the last polynomial function in the last knot  $x_n$  [12].

$$f_1''(x_0) = 0 \quad (18)$$

$$f_n''(x_n) = 0 \quad (19)$$

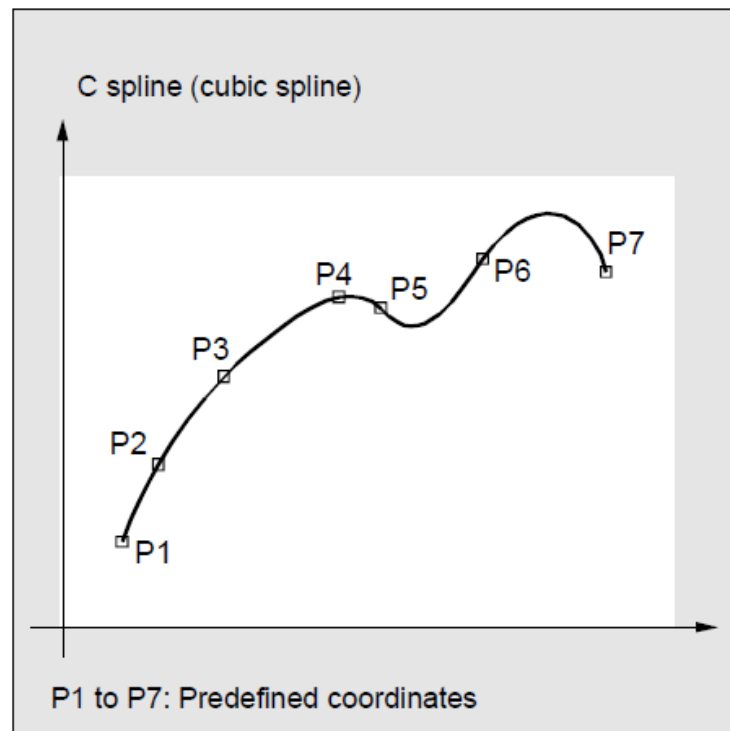


Figure 1.2.3.5 Example of C-spline interpolating 7 knots. [15]

### 1.3 NC programs

To implement machining of a workpiece the tool path needs to be generated to control respective tool-workpiece movements. NC programs are programmed following correct syntax adapted for used control system [13].

#### 1.3.1 Types of programming

We differentiate two types of programming such as absolute programming and incremental programming [4].

##### 1.3.1.1 Absolute programming

In absolute programming coordinates of all points are expressed in relation to the coordinate system origin [4]. In Sinumerik this type of programming is called by function G90 [4]. When using the G90, input is absolute dimensions while all data coordinates refers to the workpiece zero [20]. While using absolute programming the diameter of the piece is determined in axis X and its lengths in axis Z [4].

##### 1.3.1.2 Incremental programming

In incremental programming coordinates of all points are expressed in relation to the last point's coordinates [4]. In Sinumerik this type of programming is called by function G91 [5]. After calling incremental programming by function G91, all input is in incremental dimensions while each dimension refers to the contour's point last input [20].

### 1.3.2 Structure of the program

The structure of NC programs follows syntax specific to the control system of the machine. Most of control systems use similar so called G programming language that has many varieties but the most commonly used one is ISO language. Every command is composed of a letter address and its numerical value. In general it consists of letter addresses and numerical values. For better orientation in the programs, blocks (lines) are assigned a block line number consisting of letter address N and number of the block. Blocks contain commands and are executed in numerical order. The letter address can be any letter of the alphabet but the most used are G and M. Every command might or might not have a numerical value assigned to it. In Sinumerik control systems G functions are also called preparatory functions and M functions can also be called auxiliary or Miscellaneous functions. The functions and their utilisation vary for turning and for milling. While Sinumerik 840D offers advanced forms of programming like multi-channel programming, the basic programming language it uses is ISO-Code. Next passage states a few principal functions in ISO dialect mode used in its G-code according to norm DIN66025 [5], [21].

#### 1.3.2.1 Letter addresses

Table 1.3.2.1 Letter addresses [7] [18].

A	Absolute or incremental position in rotational axis A
B	Absolute or incremental position in rotational axis B
C	Absolute or incremental position in rotational axis C
G	Preparatory functions
F	Feed rate designation
M	Miscellaneous (auxiliary) functions
N	Block (line) number
S	Speed designation (of spindle)
T	Tool selection
X	Absolute or incremental position in linear axis X
Y	Absolute or incremental position in linear axis Y
Z	Absolute or incremental position in linear axis Z



### 1.3.2.2 G functions

Table 1.3.2.2 G functions [9], [20].

G0	Rapid traverse motion
G1	Linear interpolation
G2	Clockwise circular interpolation
G3	Counterclockwise circular interpolation
G4	Dwell time preset
G9	Exact stopping before continuing to the next block, diminution of velocity
G17	X/Y plane selection for linear and circular interpolation
G18	X/Z plane selection for linear and circular interpolation
G19	Y/Z plane selection for linear and circular interpolation
G33	Threading with constant lead
G40	No tool radius compensation
G41	Tool radius compensation left of the workpiece contour
G42	Tool radius compensation right of the workpiece contour
G53	Frame suppression (in concerned block)
G54	First settable zero offset
G55	Second settable zero offset
G56	Third settable zero offset
G57	Fourth settable zero offset
G70	Input in inches
G71	Input in metric system
G74	Approaching reference point
G75	Approaching fixed point
G90	Absolute dimension input
G91	Incremental dimension input
G93	Inverse time federate coding
G94	Linear feed rate Fv in mm/min or inch/min
G95	Revolution feed rate Fv in mm/revolution or inch/revolution
G96	Constant velocity on
G97	Constant velocity off

### 1.3.2.3 M functions

Table 1.3.2.3 M functions [9], [20].

M0	Programmed stop
M1	Optional stop
M2	Main program and return to program beginning
M3	Spindle turning clockwise
M4	Spindle turning counterclockwise (CCW).
M5	Spindle stop
M6	Tool change
M17	Subroutine end
M30	Subroutine end and return to program beginning
M41-M45	Gear stage 1-5

## 1.4 Spline interpolation in different control systems

Today all control systems support linear interpolation and some form of circular interpolation. If we intend to use spline interpolation, firstly we must verify if the control system supports spline interpolation [13].

Next section treats interpolation options in three different control systems: Sinumerik, Heidenhain and Fanuc.

### 1.4.1 Sinumerik

Sinumerik control systems produced by corporate company Siemens AG offer multiple spline interpolation options. Sinumerik 840 control system is the most adapted to elaborate spline interpolation and other smooth curves interpolation than other Sinumerik control systems. Sinumerik 840 offers directly A-spline, B-spline and C-spline interpolation programming. It is essential to point out that what the producer refers to as a B-spline is in reality a NURBS [13].

The introduction of all three types of splines is similar.

General syntax for programing spline interpolation in Sinumerik line solution system [9]:

ASPLINE X... Y... Z... A... B... C...

BSPLINE X... Y... Z... A... B... C...

CSPLINE X... Y... Z... A... B... C...

Abbreviations used in this syntax refer to terms in Table 1.4.1.1 [9].

Table 1.4.1.1 Spline syntax parameters in Sinumerik control system.

ASPLINE	introduction of A-spline into the program
BSPLINE	introduction of B-spline into the program
CSPLINE	introduction of C-spline into the program
X, Y, Z, A, B, C	coordinates of knots in Cartesian coordinate system
...	values to be designate by the programmer

#### 1.4.1.1 Programming B-splines in Sinumerik

For B-splines we can also program other parameters [9]:

PW=<n>

SD=2

PL=<value>

Where abbreviations have following meaning stated in Table 1.4.1.2 [9].

Table 1.4.1.2 Additional parameters for B-spline knots in Sinumerik.

PW	parameter of weight of each knot
SD	degree of the curve
PL	parameter of length between two knots

The syntax for writing a NURBS interpolation is following [13]:

BSPLINE SD=... X... Y... Z... F...

X... Y... Z... PL=... PW=...

X... Y... Z... PL=... PW=...

Abbreviations used in this syntax refer to following terms in Table 1.4.1.3 [13]:

Table 1.4.1.3 Abbreviations in syntax for NURBS spline interpolation in Sinumerik

BSPLINE	introduction of B-spline into the program
SD	degree of the curve
...	numbers to be designate by the programmer
X, Y, Z	coordinates of knots in X,Y,Z axes
F	feed-rate specification
PL	parameter of length between two knots
PW = <n>	parameter of weight of each knot

PW, SD and PL parameters are useable only for B splines and would not influence other types if spline interpolation [9], [13].

PW = <n> represents the weight of every knot. It's value can range from n= 0 to n=3 with step of 0,0001. If  $n > 1$ , the curve is pulled significantly towards the knot in question. On the other hand if  $n < 1$  the curve is pulled towards the knot slightly.

Distance between two knots is calculated automatically for optimal results but can also be changed by the programmer using the PL parameter [13].

Sinumerik 840 also offers 3D NURBS interpolation meaning it uses multiple axes. This is done by simultaneously programming two spatial curves. The first curve controls the tool reference point. The second curve controls a second point distanced by constant distance from the tool reference point. [13].

The syntax for writing a NURBS interpolation for multi-axis is following [13]:

BSPLINE SD=... F...

X... Y... Z... XH=... YH=... ZH=... PL=...

X... Y... Z... XH=... YH=... ZH=... PL=...

While abbreviations used in this syntax refer to terms in

Table 1.4.1.4 Coordinated syntax for multi-axis NURBS in Sinumerik.

X, Y, Z	coordinates of the data points of the first curve in X, Y, Z axes
XH, YH, ZH	coordinates of data points of the second spatial curve

#### 1.4.1.2 Programming A-spline and C-spline in Sinumerik

Syntaxes for programming A-spline and C-spline are similar in Sinumerik 840 control system.

After introducing the splines it the program by commands of ASPLINE or CSPLINE and entering the knot coordinates as described above, additional parameters can be added [13].

Parameters controlling transition at the beginning of spline block are stated in Table 1.4.1.5 [9].

Table 1.4.1.5 Parameters for transition at the beginning of new spline block.

BAUTO	no information about the transition compartment, beginning of the curve is determined by the first knot
BNAT	transition with zero curvature
BTAN	tangential transition

Parameters controlling transition at the end of a spline block and their meaning are stated in Table 1.4.1.6 [9].

Table 1.4.1.6 Parameters for transition at the end of last spline block.

EAUTO	no information about the transition compartment, beginning of the
ENAT	transition with zero curvature
ETAN	tangential transition

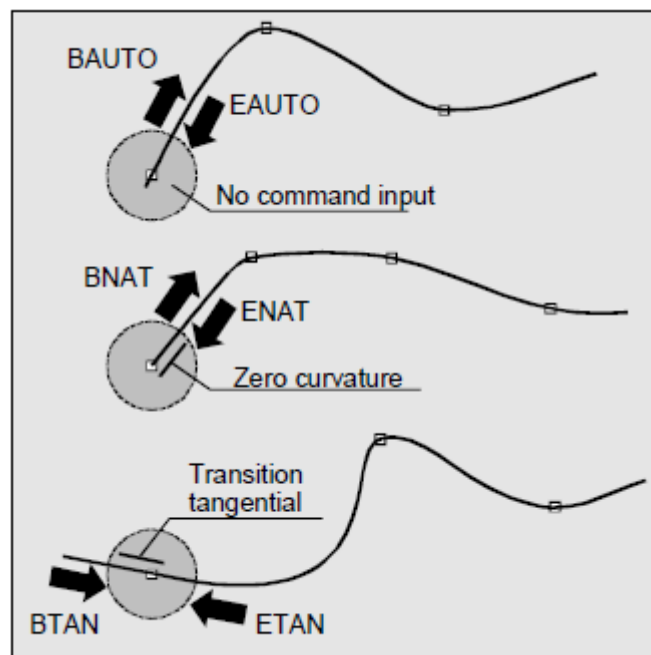


Figure 1.4.1.1 Representation of transition between blocks of spline applicable for A-spline and C-spline [15].

Entering these transition parameters into the B-spline program syntax won't result in malfunctioning or won't generate an error code but it won't affect the B-spline curve in any way [9].

Sinumerik 840 also provides choices of linear, circular and polynomial interpolation. Polynomial interpolation can be used to machine diverse curves that are approximated by a polynomial including conic sections or power functions [9].

Once an interpolation method is called, the point's coordinates that follow will be taken as knots for the interpolation. Calling another interpolation method will cancel previous interpolation method.

#### 1.4.2 Heidenhain

Even though the control systems iTNC produced by Heidenhain also offer smooth curve programming, they don't directly offer specific spline interpolation like Sinumerik. In the following section the case of iTNC530 is going to be detailed.

The splines can be programmed using the general form of third degree polynomials. iTNC offers to transfer spline from CAD systems in two, three, four or five axes polynomials [22].

Notation of polynomials in X, Y, Z axes is described in (20) [13].

$$\begin{aligned} X(t) &= K3X \times t^3 + K2X \times t^2 + K1X \times t + X \\ Y(t) &= K3Y \times t^3 + K2Y \times t^2 + K1Y \times t + Y \\ Z(t) &= K3Z \times t^3 + K2Z \times t^2 + K1Z \times t + Z \end{aligned} \quad (20)$$

$$t \in \langle 0; 1 \rangle$$

Notation of third degree polynomials in two rotary axes (here represented by letters A and B, but other combinations of two axes out of of axis A, B and C are possible) can be written as (21) [22].

$$\begin{aligned} A(t) &= K3A \times t^3 + K2A \times t^2 + K1A \times t + A \\ B(t) &= K3B \times t^3 + K2B \times t^2 + K1B \times t + B \end{aligned} \quad (21)$$

While abbreviations used in syntax (21) stand for terms cited in Table 1.4.2.1 [22].

Table 1.4.2.1 Abbreviations in syntax for programming third degree polynomials in Heidenhain.

X(t), Y(t), Z(t)	third degree polynomials for coordinate system axes X,Y,Z
A(t), B(t)	third degree polynomials for rotary axes A and B
X, Y, Z, A, B	coordinated of the end point
KX3, KX2, KX1	polynomial coefficients for axis X
KY3, KY2, KY1	polynomial coefficients for axis Y
KZ3, KZ2, KZ1	polynomial coefficients for axis Z
KA3, KA2, KA1	polynomial coefficients for axis A
KB3, KB2, KB1	polynomial coefficients for axis B
KB3, KB2, KB1	polynomial coefficients for axis B

Values of t vary from 0 to 1 while the step span can be modified [22].

It's important to note that spline sequences cannot be furthermore edited once it is transferred in to the iTNC control system with the exception of changing federate and

changing M functions This fact the usage of this method for both reverse engineering and machining of an analytical curve because CAD system still need to be used to generate the spline before translating it into pieces of third degree polynomials by using a post-processor [22].

### 1.4.3 Fanuc

Fanuc has also developed curve interpolation functions and NURBS interpolation for their most advanced control system [13]. Fanuc 31i control system syntax will be elaborated in the following section.

Fanuc control system offers only NURBS interpolation and uses it for so called Nano-Smoothing. G06.2 is an inbuilt function of NURBS interpolation. Other spline interpolation are not available. [23]

G functions and parameter used in Fanucs 31i for programming NURBS interpolation are stated in Table 1.4.3.1.

Table 1.4.3.1 G functions and parameters used NURBS programming in Fanuc.

G06.2	NURBS interpolation, cancelled by other motion command: G00,G01,G02 or G03
P	rank of NURBS curve 2,3 or 4 (4 by default, otherwise based on CAD model)
XYZ	control points coordinates (three or more depending on number of axis)
R	weight (by default R=1,0 if not specified)
K	Knot
F	Interpolation feedrate
G112	Polar coordinate interpolation mode
G113	Polar coordinate interpolation mode cancelling
G40	Cutter or tool nose radius compensation: cancel Three-dimensional cutter compensation: cancel

Number of specified knot points equals to number of control points plus the value of rank. Parameters weight R and knot K are both nine digit absolute value of the minimum data unit of the reference axis – in millimetres: -999999,999 to 999999,999. In Fanuc control systems, NURBS are never used in cutter radius compensation mode, G40 must be in effect BEFORE calling G06.2 [23].

## 1.5 Comparison of interpolation methods for CNC

Especially in process of finishing it's necessary to employ trajectories adapted to the morphology of the machined surface. In case of warped surfaces, linear and circular interpolations are not sufficient: while using linear interpolation, the acceleration is variable and bezels (facets) are formed on the machined surface [6]. With sufficient number of points, linear interpolation can meet set demands, but the acceleration shortcoming would still lengthen the machining time. Circular interpolation isn't sufficient either because it is not always possible to fit a general smooth curve with pieces of circles especially if only a limited number of points of the curve are known. [8], [14].

To eliminate this defect more elaborate methods can be used: spline interpolation, NURBS interpolation or polynomial interpolation. These methods enable us to machine complex surfaces in less time because they are based on mathematical model of class  $C^2$  meaning the acceleration is invariable during the whole process of machining [6], [8].

That's why it's favourable to implement spline interpolation in specific cases where only limited number of points of the curve (or surface) is known.

On Figure 1.4.3.1 differences between three spline interpolations that were previously described are pictured. The main distinction of the B-spline interpolation is that the curve doesn't pass directly through the knots. In case of NURBS this disadvantage can be reduced by adding weight to each knot to pull the curve closer to the knot, yet it will never pass through the points either. This can cause not sufficient geometrical precision when the number of points is not sufficient.

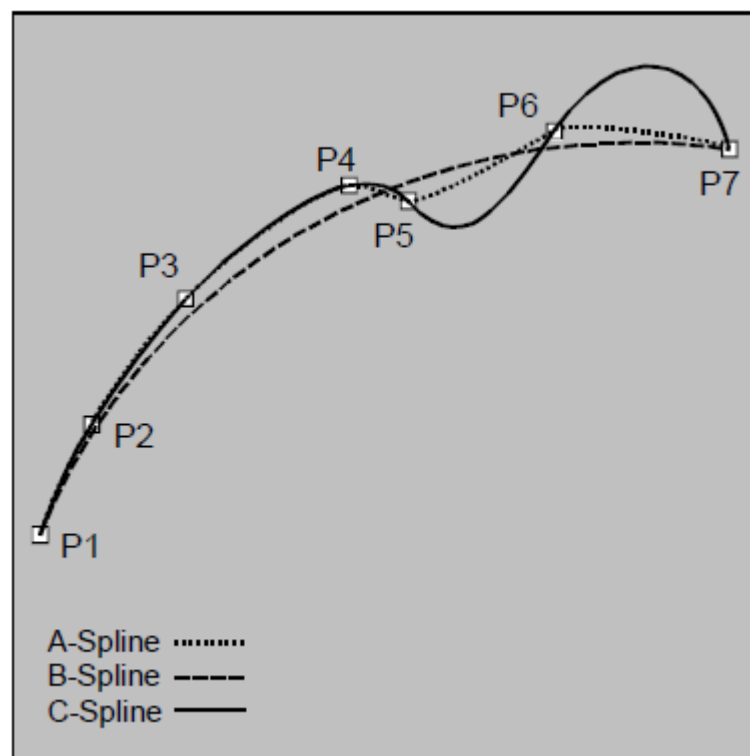


Figure 1.4.3.1 Comparison of A-spline, B-spline and C-spline interpolating same 7 knots [15].

To further develop practical usage of spline interpolation for CNC machining and limits of each method, a series of experiments was designed and executed as described in the following sections.



## **2 DESIGN OF A CNC PROGRAM USING SPLINE INTERPOLATION**

Milling of an analytically describable contour was chosen in order to compare interpolation methods described in the theoretical analysis.

The application aims to compare theoretical curve with a machined curved obtain by machining using different interpolation methods while changing number of knots on the curve length and analyse its influence.

Sinumerik 840 control system was chosen out of three control system studied in (1.4) as the most suitable to conduct the practical study of spline interpolation because it offers directly A-spline, B-spline and C-spline interpolation syntax. Sinumerik 840 control system was employed at Brno University of Technology technological on its five-axis CNC milling machine TAJMAC-ZPS MCV 1210 therefore it was possible to execute the application there.

To test spline interpolation methods, knots on an analytically know curve needed to be generated. A contour was generated in MATLAB programming platform and knots were symmetrically placed on the contour for two types of knot distribution: polar and equidistant. The knots were then implemented in subprograms after calling corresponding method to interpolate the knots in the main program.

The main programs were written directly on the CNC milling machine. The subprograms containing only the knot's coordinates were created in MATLAB and imported to the machine's control system memory.

Interpolation methods were tested on both 2D and 3D closed contour.

MATLAB programming platform was used to generate coordinates of the knots based on an analytically known contour. In this programming platform it was possible program multiple scripts to generate knots with different parameters and have the script write their coordinates in correct syntax directly into MPF files readable by the machine. These MPF files were then used as subprograms. Various figures were also generated to visualise the knots on the theoretical curve. Versions 2017b and 2018a of the platform were used, some of their features aren't available in older versions.

### **2.1 Creation of the contour**

Experimental analytically known contour was chosen according to the following main conditions:

- the curve must be closed in order to create a contour,
- the origin of correspondent Cartesian coordinate system must be situated in the centre of the contour,
- the contour must have at least two axes of symmetry. In case of a closed contour, problems with connection where the beginning and the end of the contour meet were expected. That's it was important to have a symmetrical contour in order to be able to compare the part influenced by beginning and to a part beyond its influence,

- the contour must have local maxima of different distance from origin  $r_i$ .

Creation of an analytically describable symmetrical contour corresponding to previous conditions was set as a goal. Polar curves with equation constituting of trigonometrical functions sine and cosine and their multiplications or additions match this goal perfectly as is going to be further elaborated. The search for an equation of a suitable polar curve followed.

### 2.1.1 Curve in polar coordinate system

A polar curve is constructed thanks to its equation in the polar coordinate system. Coordinates of point A in polar coordinate system are defined by its distance from the origin O called  $r_A$  ( meaning  $r_A = (OA)$ ) and angle  $\theta_A$  between  $r_A$  and one of the axes (most commonly axis X of corresponding Cartesian coordinate system) as is displayed at figure Figure 2.1.1.1 [24].

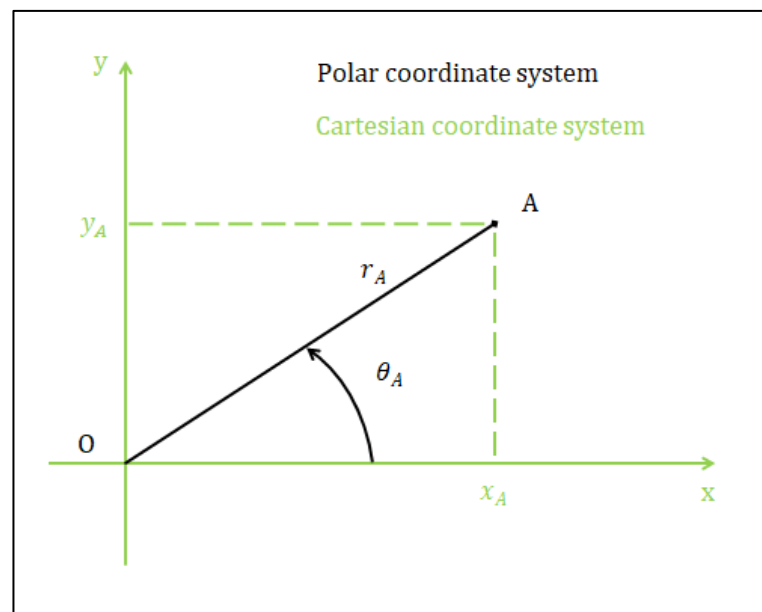


Figure 2.1.1.1 Polar and Cartesian coordinate systems.

Transformation of polar coordinates to Cartesian coordinates is thanks to trigonometrical functions sine and cosine of angle  $\varphi_A$  by equalling (22) [24].:

$$\begin{aligned} x_A &= r_A \cos(\theta_A) \\ y_A &= r_A \sin(\theta_A) \end{aligned} \quad (22)$$

All points of a polar curve are defined by variable distance from the origin  $r$  as a function of angle  $\theta$ . A polar curve is therefore defined by an equation of a distance  $r$  that is a function of angle,  $r = f(\theta)$ . [24]

MATLAB 2017b and 2018a offer function *polarplot* to generate a representation polar curves. Polarplot's parameters are equation of the distance from the origin  $r$  and a vector theta.

Simpler function called *plot* in older versions of the software can be also used while explicitly posing  $x = r\cos(\theta)$  and  $y = r\sin(\theta)$  and plotting  $x$  and  $y$  coordinates in Cartesian system. Both functions were used when choosing an equation of the contour.

### 2.1.2 Choosing an equation of the contour

To create a closed contour with origin in the middle of the contour it is necessary that theta varies in total of  $2\pi$  radians. If it starts for theta  $\theta = 0$  radians it therefore should vary from 0 to  $2\pi$ , polar curves with equation constituting of trigonometrical functions will create closed contour because of their periodicity based on certain angle. Sine and cosine values vary both from -1 to 1. It is therefore important to add an invariable superior to absolute value of minimal value of the trigonometrical function if said minimum is negative, in order to create a curve that will not pass through the origin of the coordinate system and the value of  $r$  is always positive as illustrated on Figure 2.1.2.1.

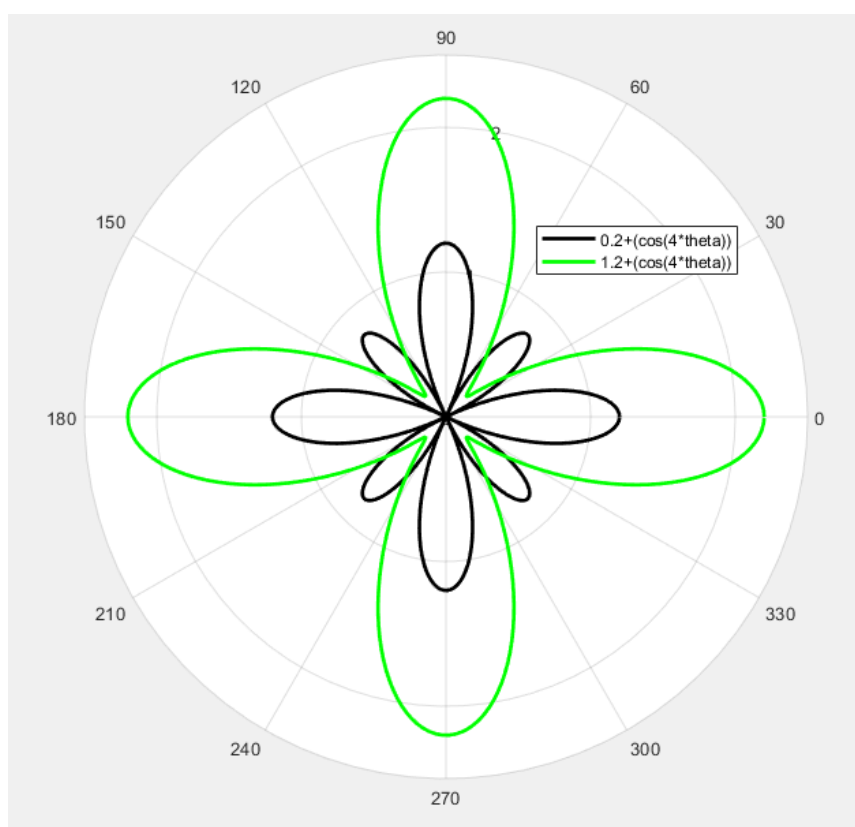


Figure 2.1.2.1 Curve not passing through origin creating a closed contour

Sine and cosine have period of  $2\pi$ , while tangent and cotangent have period of  $\pi$ . Nevertheless tangents and cotangent are not continuous functions, therefore their polar representation does not conclude in a symmetrical contour. Tangents and cotangents are by definition (23), (24).

$$\tan = \frac{\sin(\theta)}{\cos(\theta)} \quad (23)$$

$$\cotan = \frac{\cos(\theta)}{\sin(\theta)} \quad (24)$$

Any fragments including fragments of sine and cosine in the denominator will cause asymmetry of the polar curve, because of possible division by 0 for angles theta that results in null sine or cosine and will resolve in discontinuity of the curve as represented on Figure 2.1.2.2. This includes functions tangent and cotangent. The equation of this contour contains division by cosine of theta. As is well visible, the value  $\cos(\theta = 90^\circ) = 0$  causes that limit  $\lim(1/\cos(\theta = 90^\circ = \pi/2 \text{ rad}))$  is approaching the value of infinity, causing a defect in that area. The same goes for value of theta  $\theta = 270^\circ = 3\pi/2 \text{ rad}$

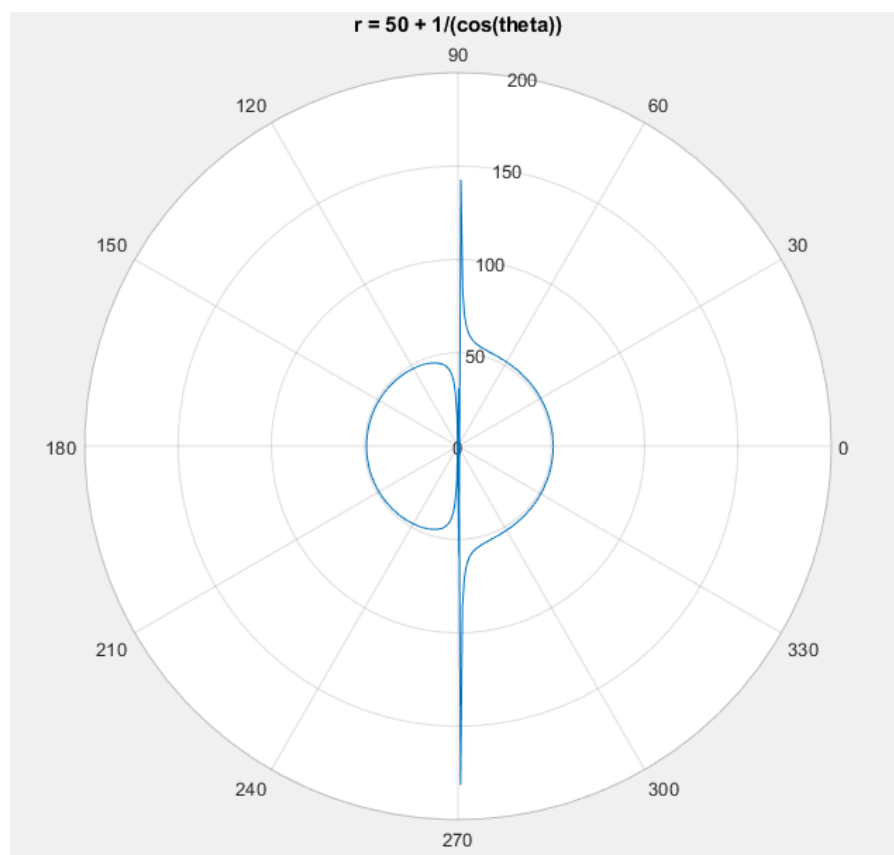


Figure 2.1.2.2 Influence of including fragment with sine or cosine in the denominator in equation of on the polar curve.

Using only sine and cosine and their additions or multiplications also ensures to symmetry of the contour as long as the equation does not contain any other variable containing  $\theta$  as illustrated in Figure 2.1.2.3.

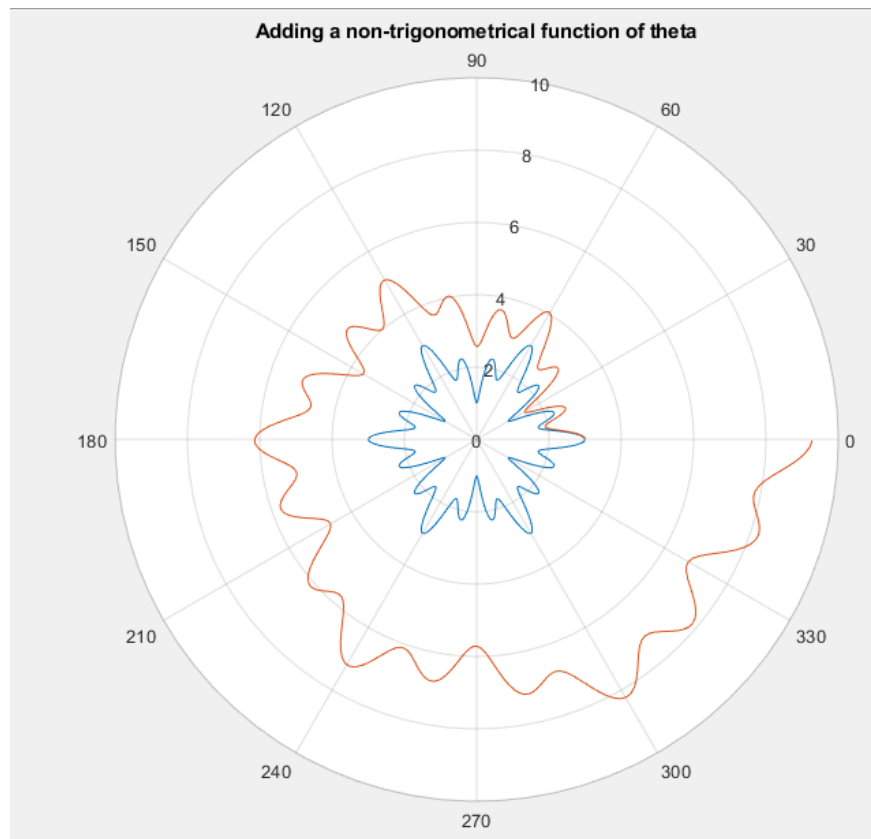


Figure 2.1.2.3 Deformed figure after adding a non-trigonometrical function of theta.

The blue curve has polar equation (25). While the red curve has for equation (26).

$$\text{contour} = 2 + \cos(12\theta) * \cos(6\theta) \quad (25)$$

$$\text{curve} = 2 + \cos(12\theta) * \cos(6 * \theta) + \theta \quad (26)$$

The only difference between the two equations (25) and (26) is the addition of constituent  $+\theta$  in (26), causing the curve to gradually increase the value of  $r_i$  with increasing value of theta and therefore the contour created will not form a closed contour. However addition of the constant wouldn't result in not creating a contour, it will only enlarge the distance of all points form the origin of the coordinate system by the value of the constant.

As was stated only polar curves containing additions and multiplications of sine and cosine were acceptable for creation of a closed symmetrical contour. A general form of a polar contour that obliges all conditions previously set can be expressed as (27), where expression *trigo* stands for a function containing only cosine and sine functions of theta and their additions or multiplications and positive powers. A is a constant that does not influence the shape of the curve only its overall size as can be remarked on Figure 2.1.2.4.

$$r = A * (B + \text{trigo}) \quad (27)$$

Influence of value of invariable A on equation (28) is illustrated in Figure 2.1.2.4.

$$r_1 = A * (1.5 + \cos(16\theta) * \sin(8\theta) * \sin(8\theta)) \quad (28)$$

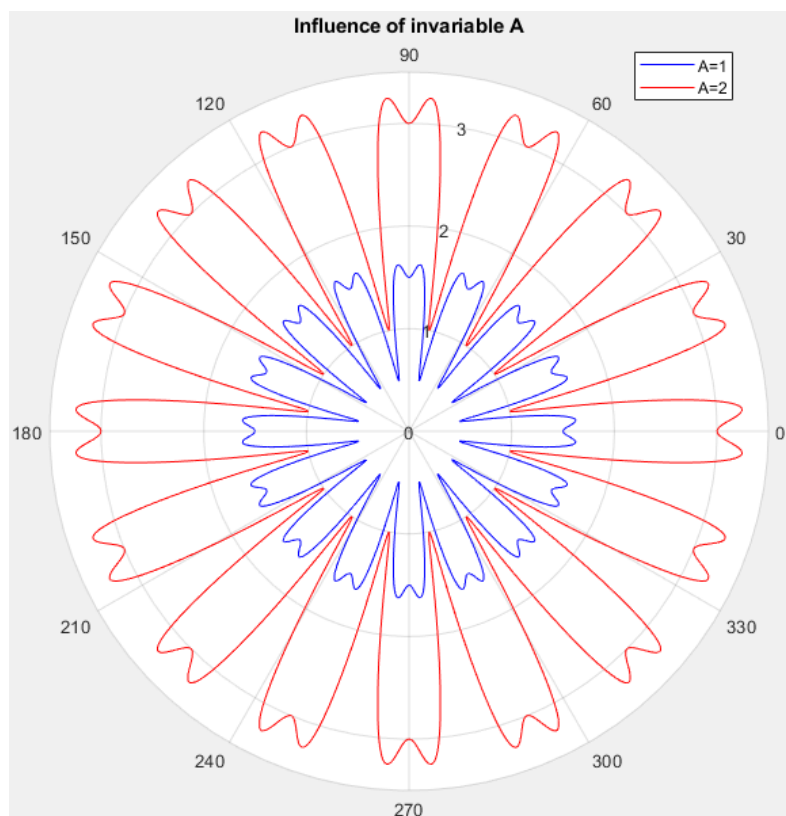


Figure 2.1.2.4 Influence of value of A.

As indicated by condition (29), B must be a constant superior to minimal value of function trigo to ensure that the curve creates a contour and doesn't pass through the origin of the polar coordinate system.

$$B > \min(\text{trigo}) \quad (29)$$

Influence of value of B is illustrated in Figure 2.1.2.5.

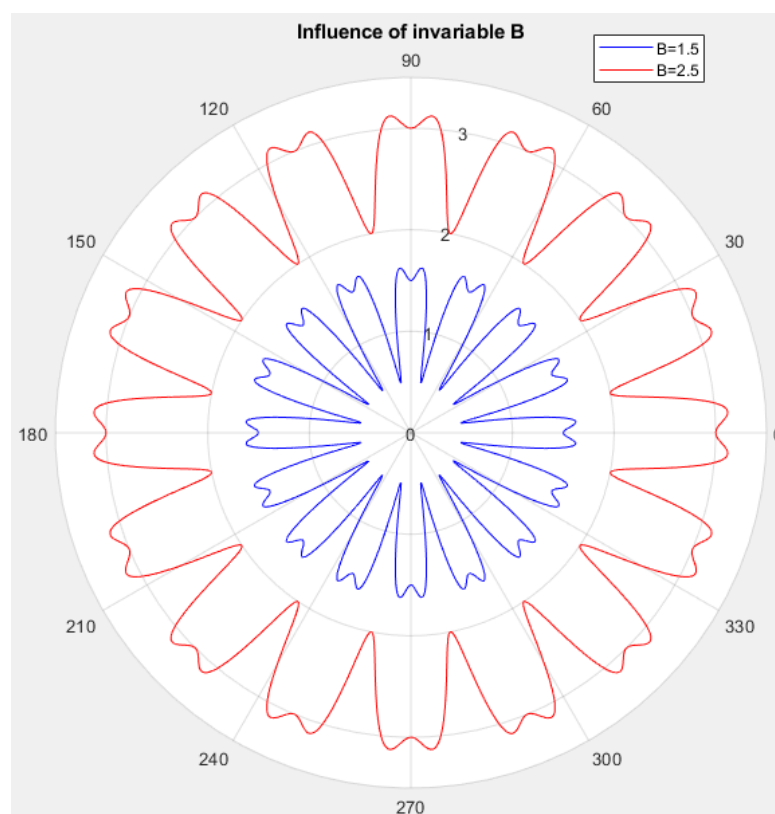


Figure 2.1.2.5 Influence of value of B.

### Axes of symmetry

As number of axes at least 2 and different local maxima were set as a condition, parameters influencing the number of axes of symmetry followed and number of local maxima followed.

After experimenting with different multiplications of theta entering the cosine and sine functions, following rules were found to be effective:

- 1) In case of functions with only multiplications of sine and cosine, the addition of all angles that appear in the trigonometrical functions divided by theta are equal to number of local maxima present in interval  $\theta \in [0; 2\pi]$

For example:

$$\text{trigo} = 1.5 + \cos(m \cdot \theta) \cdot \sin(n \cdot \theta) \cdot \cos(o \cdot \theta) \cdot \sin(p \cdot \theta)$$

$$m = 2$$

$$n = 4$$

$$o = 4$$

$$p = 2$$

$$\text{add} = m + n + p + o + q = 12$$

As addition of all variables of trigonometrical functions divided by theta is equal 12, there should be 12 local maxima present as is apparent in the Figure 2.1.2.6.

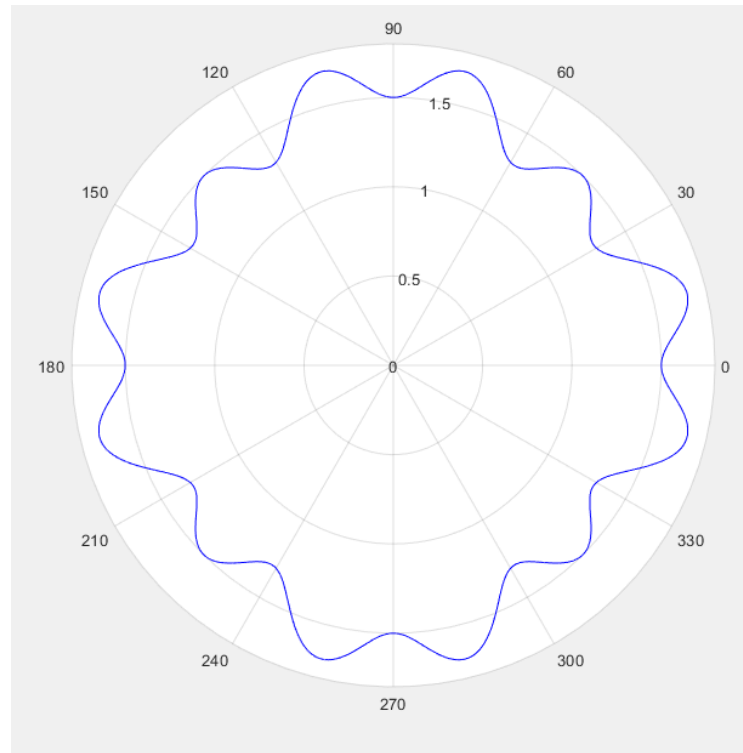


Figure 2.1.2.6 Twelve local maxima.

- 2) In case of pure multiplication case, it's also possible to predict number of axes of symmetry. If the addition of all variables of trigonometrical functions divided by theta is dividable by  $2^k$  the polar curve will present at least  $2^k$  axes of symmetry.

In figure ... the addition number is equal to 12, that is dividable by both  $2^1$  and  $2^2$ , the latter being superior the curve can be expected to have at least 4 axes of symmetry which is the case.

- 3) In case of addition, the constituent with the sine or cosine with bigger multiple of theta in the variable will dominate when it comes to number of maxima as presented following example:

$$q=4$$

$$r=8;$$

$$\text{trigo} = 5 + \cos(q*\theta) + (r*\theta);$$

As  $r > q$ , it will dictate the number of maxima, which is indeed equal 8 as visible on Figure 2.1.2.7.



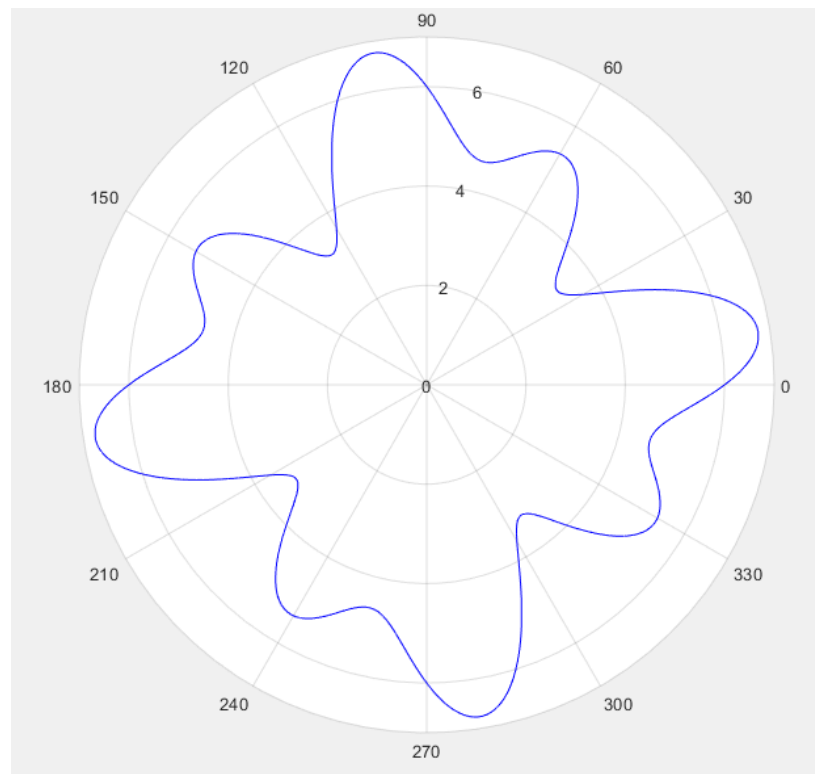


Figure 2.1.2.7 Eight local maxima.

- 4) When there is an addition of sine or cosine in the equation of the curve, it's hard to predict number of axes of symmetry.
- 5) Equations containing purely cosine functions or purely sine functions are generally symmetrical by both x and y axes of correspondent Cartesian coordinate system as illustrated in Figure 2.1.2.8.

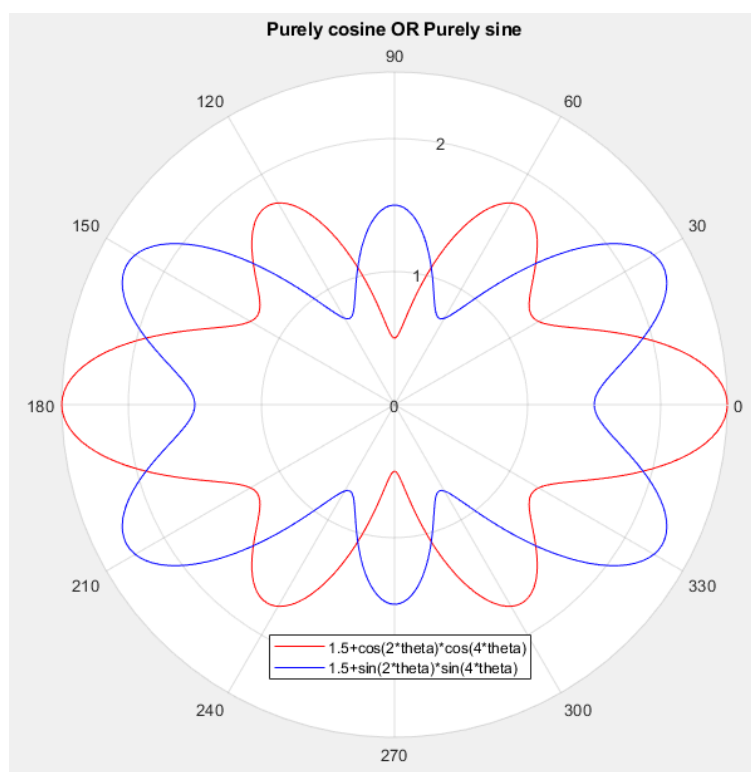


Figure 2.1.2.8 Using only sine or cosine functions in the general form

Multiple contours were generated using the findings previously described.

A trigonometrical function containing only multiplications of sine and cosines was chosen to facilitate prediction of the nature of the curves, especially its axes of symmetry. Examples of generated contours are displayed on Figure 2.1.2.9 and Figure 2.1.2.10.

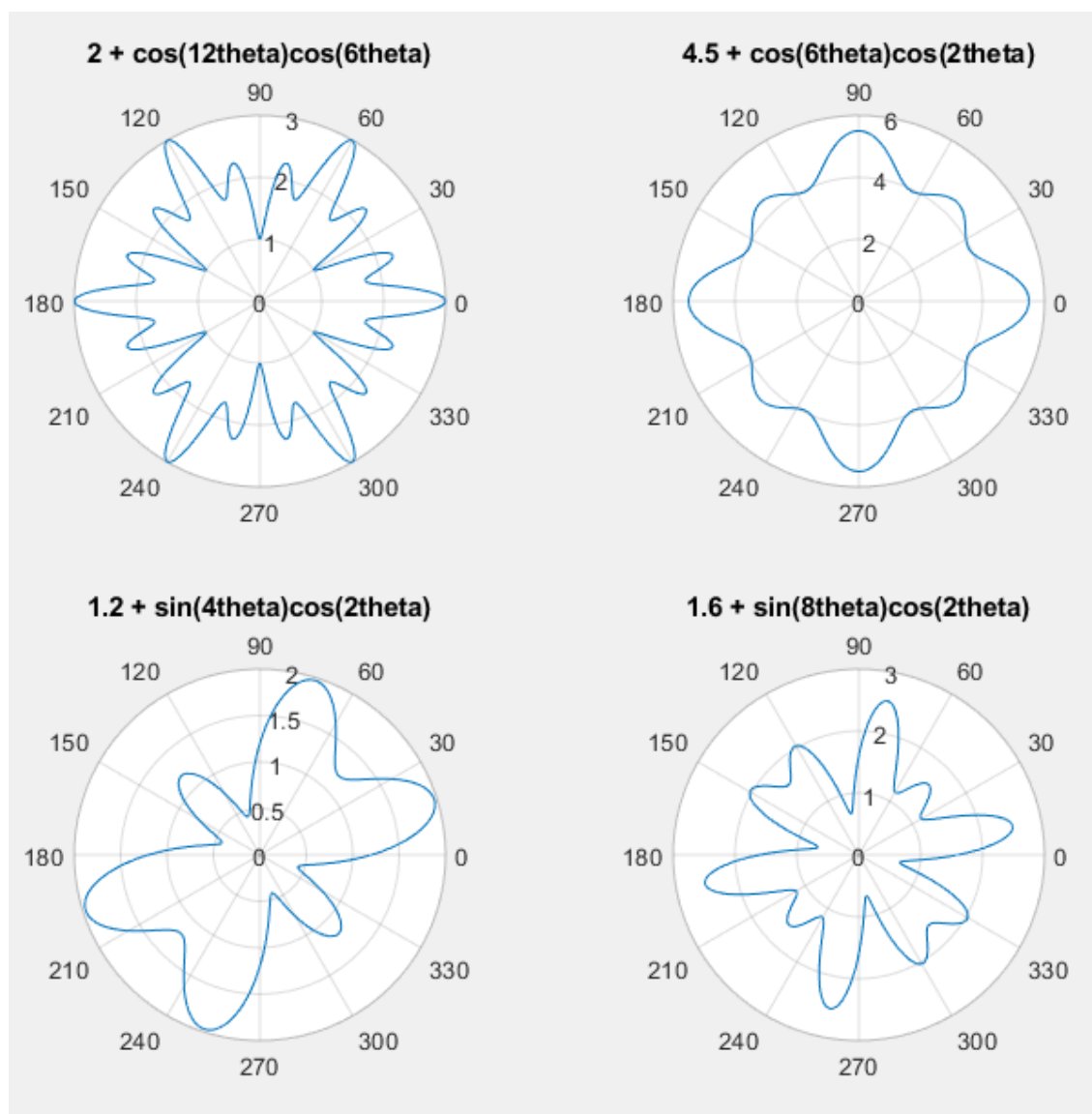


Figure 2.1.2.9 Examples of generated polar contours 1.

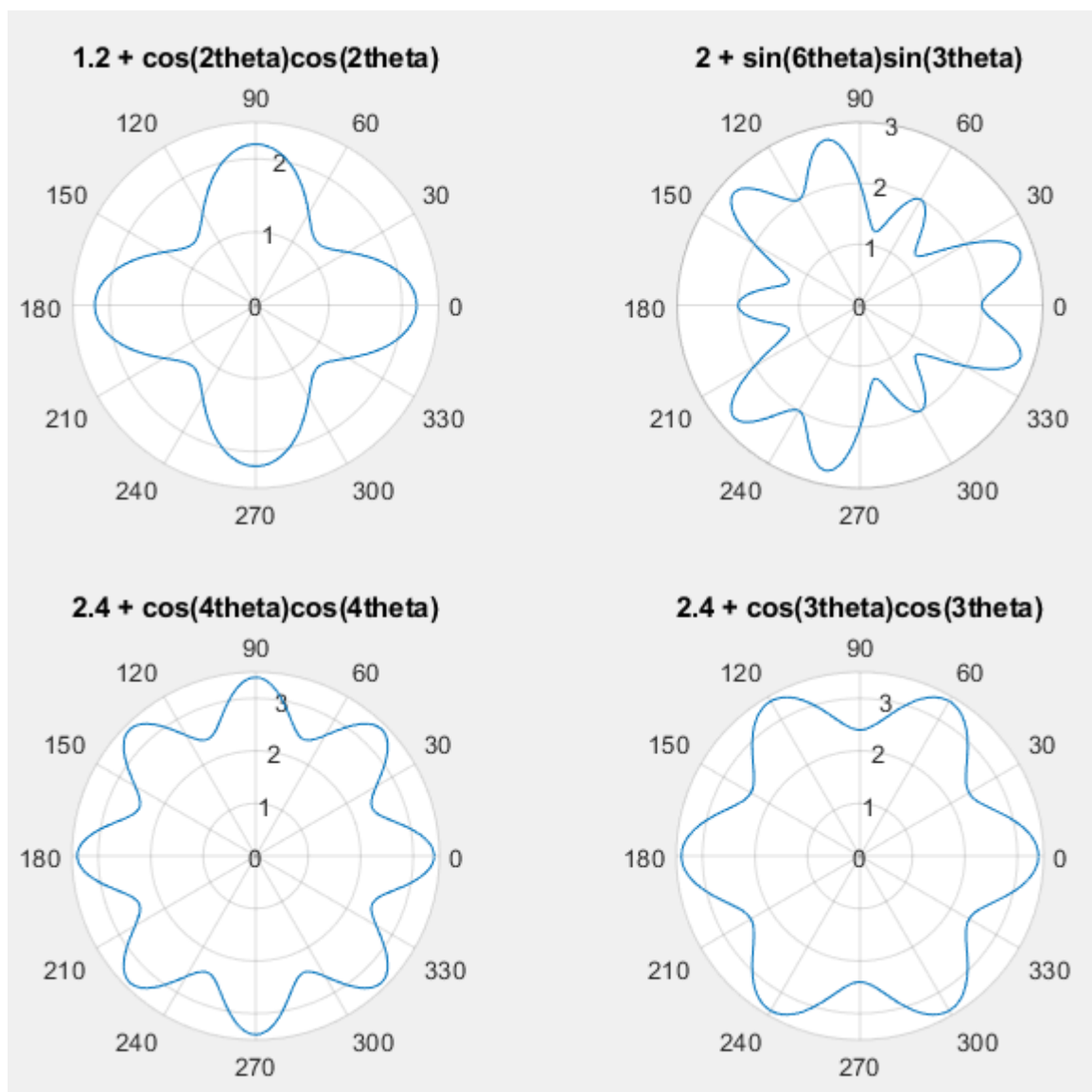


Figure 2.1.2.10 Examples of generated polar contours 2.

The curve with equation (30) was chosen as the curve to mill because it fitted the set condition the best. As can be remarked on Figure 2.1.2.11, the contour presents 4 axes of symmetry

$$4.5 + \cos(6\theta)\cos(4\theta) \quad (30)$$

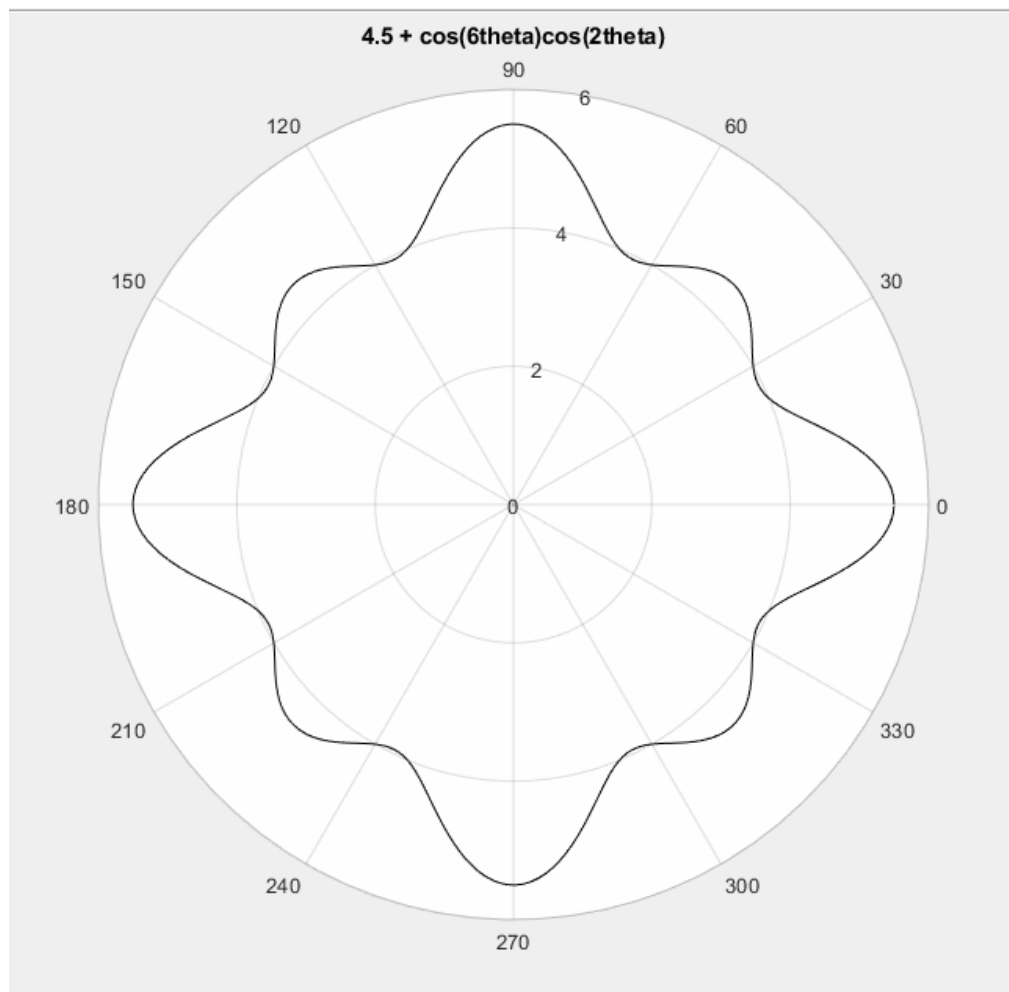


Figure 2.1.2.11 Chosen curve in polar coordinate system.

The contour was then augmented to have reasonable dimensions for milling and later analysis of results. This was done by multiplying the whole equation by 4. Then we obtain equation of the polar curve (31).

$$18 + 4 * \cos(6\theta) * \cos(4\theta) \quad (31)$$

How the figure fits the workpiece is displayed on Figure 2.1.2.12.

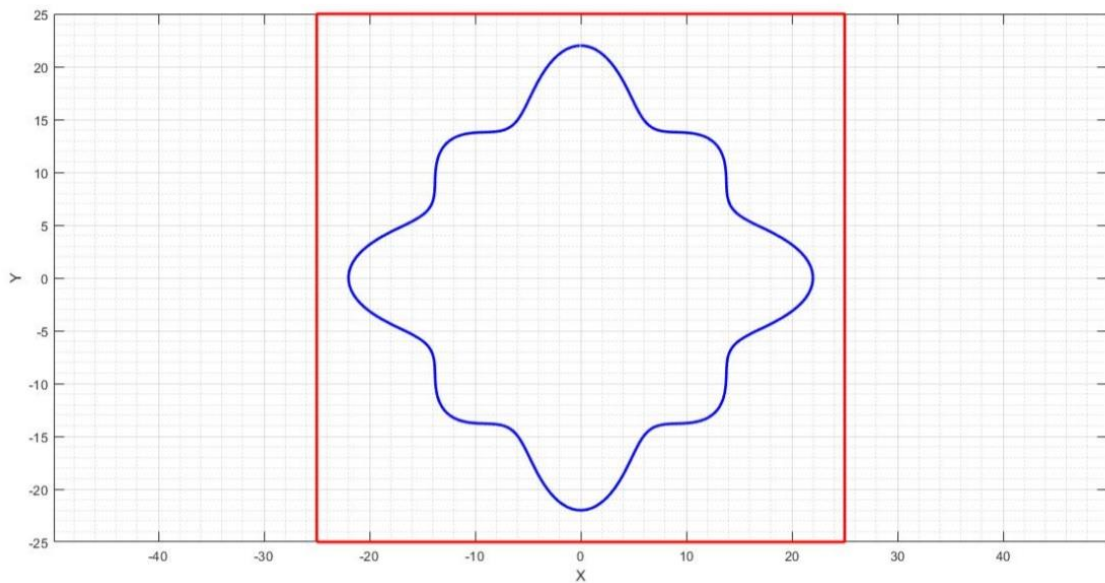


Figure 2.1.2.12 Contour on the workpiece in Cartesian coordinate system

### 2.1.3 Arc length

Arc length of the contour was calculated using a general formula for arc length (32) [25].

$$L = \int_a^b \sqrt{r^2 + \left(\frac{dr}{d\theta}\right)^2} d\theta \quad (32)$$

For the chosen contour it means calculating following integral (33).

$$\int_0^{2\pi} \sqrt{(18 + 4 \cos(2\theta) \cos(6\theta))^2 + (-8 \cos(6\theta) \sin(2\theta) - 24 \cos(2\theta) \sin(6\theta))^2} d\theta \quad (33)$$

The length of the curve was calculated to be 136,1895 mm, which was verified by MATLAB function arclength© created by John D'Errico.

### 2.1.4 Minimal radius of curvature

To ensure that the chosen curve would be machinable with standard diameters of tools and the machining would be realisable by one tool only, the radius of curvature of the contour was studied. Its minimal value was decisive as radius of the tool must be inferior to the minimal radius of curvature.

Rayon of curvature  $\rho$  is by definition reciprocal value of said curvature (34) [26].

$$\rho = \frac{1}{K} \quad (34)$$

Geometrical representation of curvature is displayed on Figure 2.1.4.1.

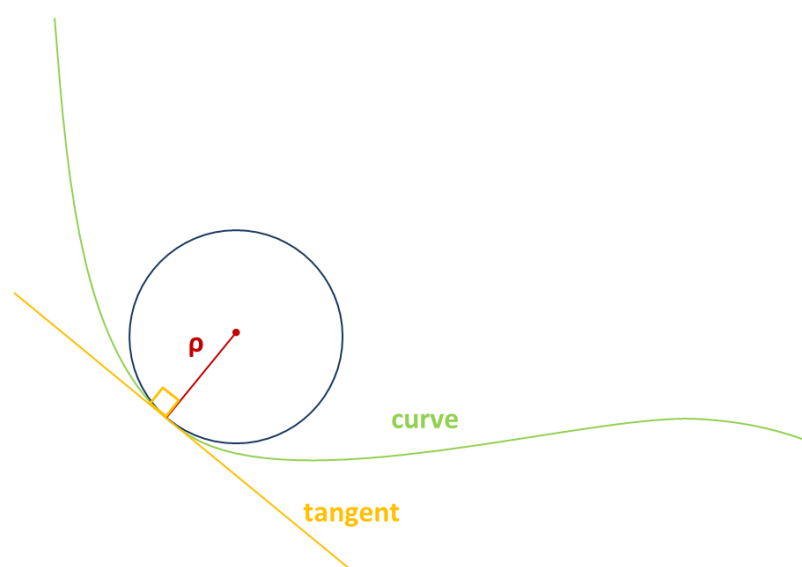


Figure 2.1.4.1 Representation of radius of curvature

For a curve with parametric equation in Cartesian coordinates, the curvature is calculated by (35) [26].

$$K = \frac{|x'y'' - y'x''|}{(x'^2 + y'^2)^{3/2}} \quad (35)$$

The equation (34) was employed in MATLAB script called `radius_of_curvature` available in annexe. The minimal rayon of curvature was calculated  $r_{\min} = 2.2966$  mm. The rayon of the tool used must therefore be inferior to this value and it was chosen to be  $r_{\text{tool}} = 2$  mm.

## 2.2 Generating knots

Once the theoretical curve was determined, knots needed to be generated. Their coordinates were then printed in .MFP files using the correct syntax for Sinumerik control system described in 1.4.1.

Multiple distributions of knots are possible on a curve, for example [27]:

- distribution with constant arc length between every two knots, represented by `dl` on Figure 2.1.4.1,
- distribution with constant angle of tangents, represented by `da` on Figure 2.1.4.1,
- distribution with constant second power of derivations of each section, represented by `df` on Figure 2.1.4.1.

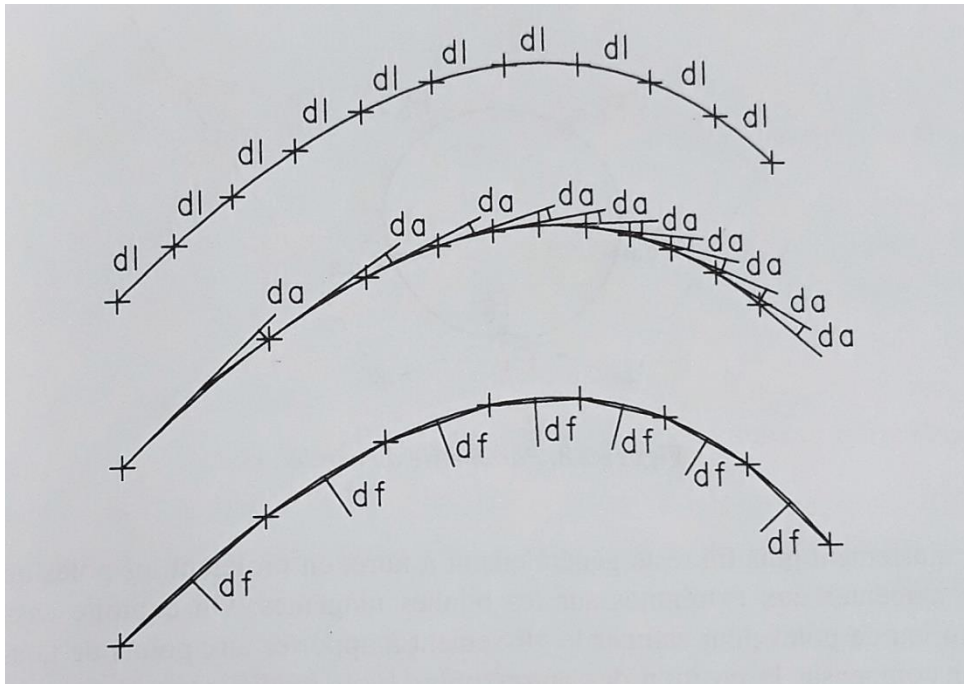


Figure 2.1.4.1 Different distribution of points on a contour [27].

It was decided to compare two types of distribution of points: polar distribution and equidistant distribution of knots. Three variations of polar distribution and three variations of equidistant distribution of knots were created in order to study influence the number of knots and the length between them on the accuracy of a chosen spline interpolation method.

### 2.2.1 Polar distribution

As was elaborated in 1.1.4, most CNC machines work with Cartesian coordinate system. The curve was generated in a polar coordinate system that is why the coordinates needed to be transformed using relations (21) between the two systems as described in 2.1.1. This relation was employed in MATLAB scripts to generate knots, but as the knots were originally obtained from a polar equation they weren't equidistantly spaced

The knots were distributed on the curve based on the step by with the value of theta advanced from 0 to  $2\pi$  in the polar equation of the curve. Smallest step of theta employed was set at  $1^\circ = \pi/180$  rad.

The number of points on the curve was therefore dependent on value of step theta. By changing its value three series of knots were generated:

- first variation had maximal number of knots, with a knot every  $1^\circ$ , meaning total of 360 points were generated on interval,
- second variation had a knot places on the curve every  $5^\circ$  with total knot count equal to  $360/5 = 72$  knots,
- third variation had minimum number of knots, with a knot placed every  $15^\circ$  and number of points  $360/15 = 24$  knots.



Table 2.2.1.1 Number of knots for three variations of polar distribution

Parameter = steptheta	1°	5°	15°
number of knot per curve	360	72	24

The values of step theta were chosen in order to keep the knot distribution symmetrical by both x and y axis. That posed a condition (36) which must have been met.

$$\frac{90}{steptheta} \in Z \quad (36)$$

Polar distribution of knots is represented for all three variations of values of steptheta in Figure 2.2.1.1, Figure 2.2.1.2 and Figure 2.2.1.3.

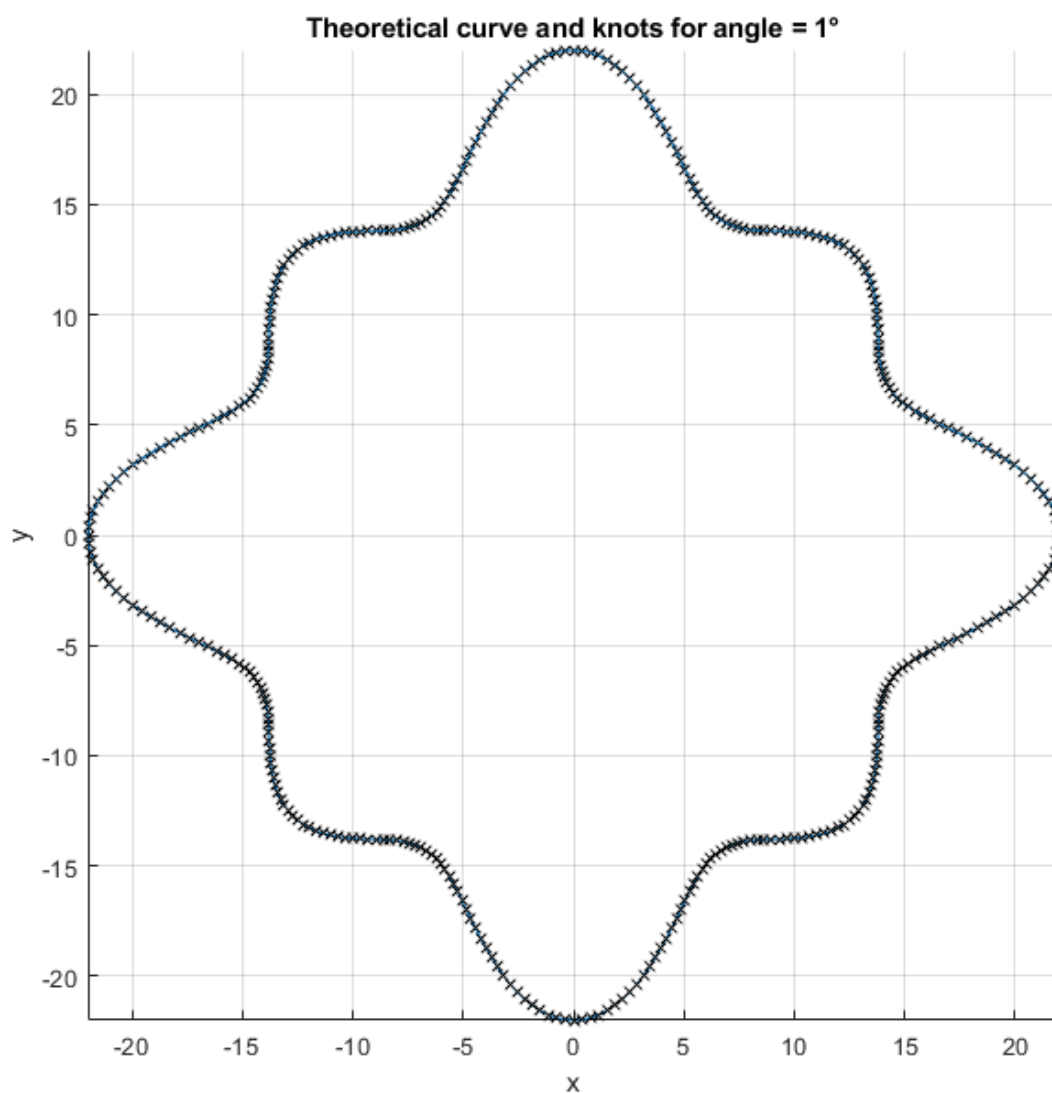


Figure 2.2.1.1 Polar distribution of knot for steptheta 1°.

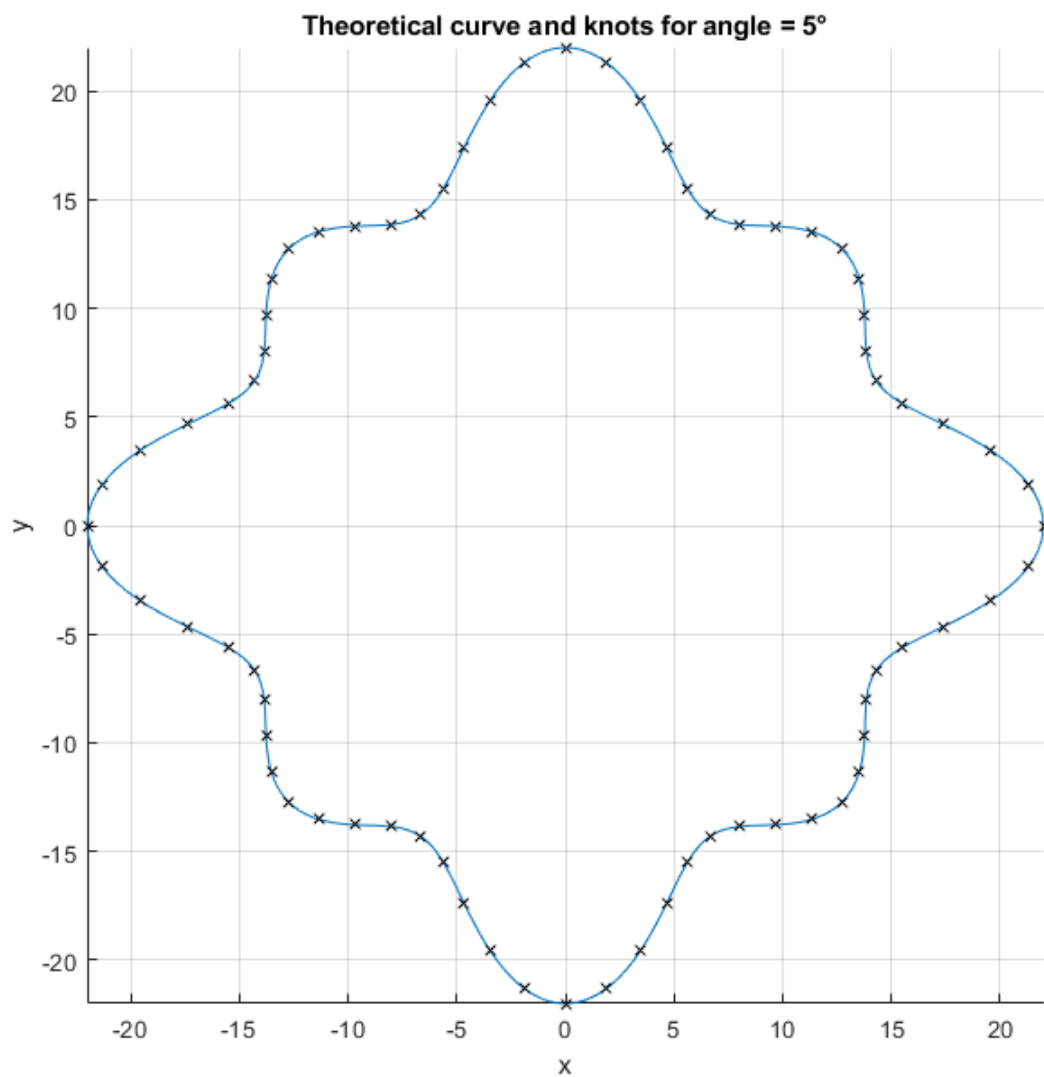


Figure 2.2.1.2 Polar distribution of knot for steptheta  $5^\circ$ .

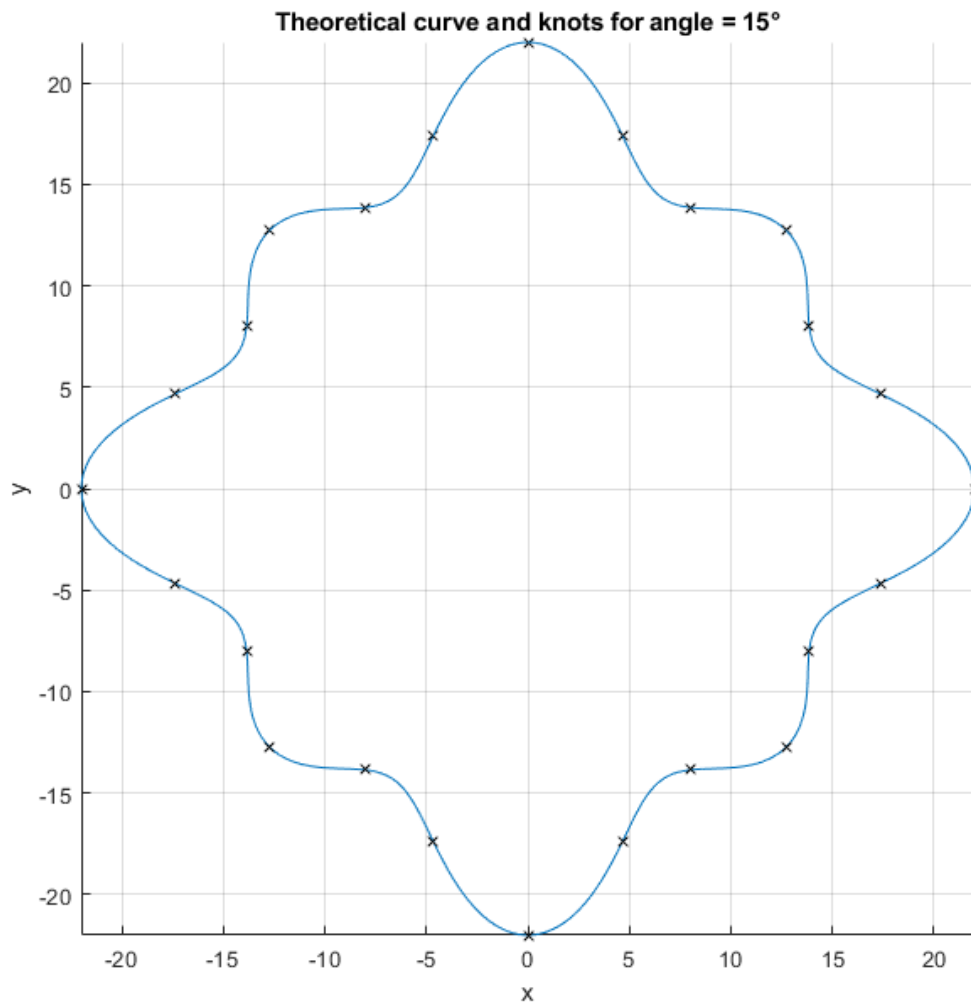


Figure 2.2.1.3 Polar distribution of knot for steptheta 15°.

### 2.2.2 Equidistant distribution

Second type of distribution employed was equidistant repartition of knots along the contour meaning the distance between knots is constant. To generate equidistant points on the curve in Cartesian (rectangular) coordinate system  $(O, x, y)$  a MATLAB function `interparc` created by John D'Errico. This function was implemented in the scripts to generate knots for the equidistant type of distribution of knots. This function needs an input of point on the curve and then employs interpolation methods to generate equidistant points on the curve. It was possible to provide a lot of polar points as the equation of the curve was known and it was possible to generate great number of points by entering small value of step theta. Therefore the interpolation was very precise and we can use output of `interparc` function as points on the theoretical contour. The function `interparc` was called on a section of the curve in the first quadrant and thanks to symmetry of the curve and the

origin of the coordinate system, the x and y knot coordinates had the same absolute value but different sign in three other quadrants.

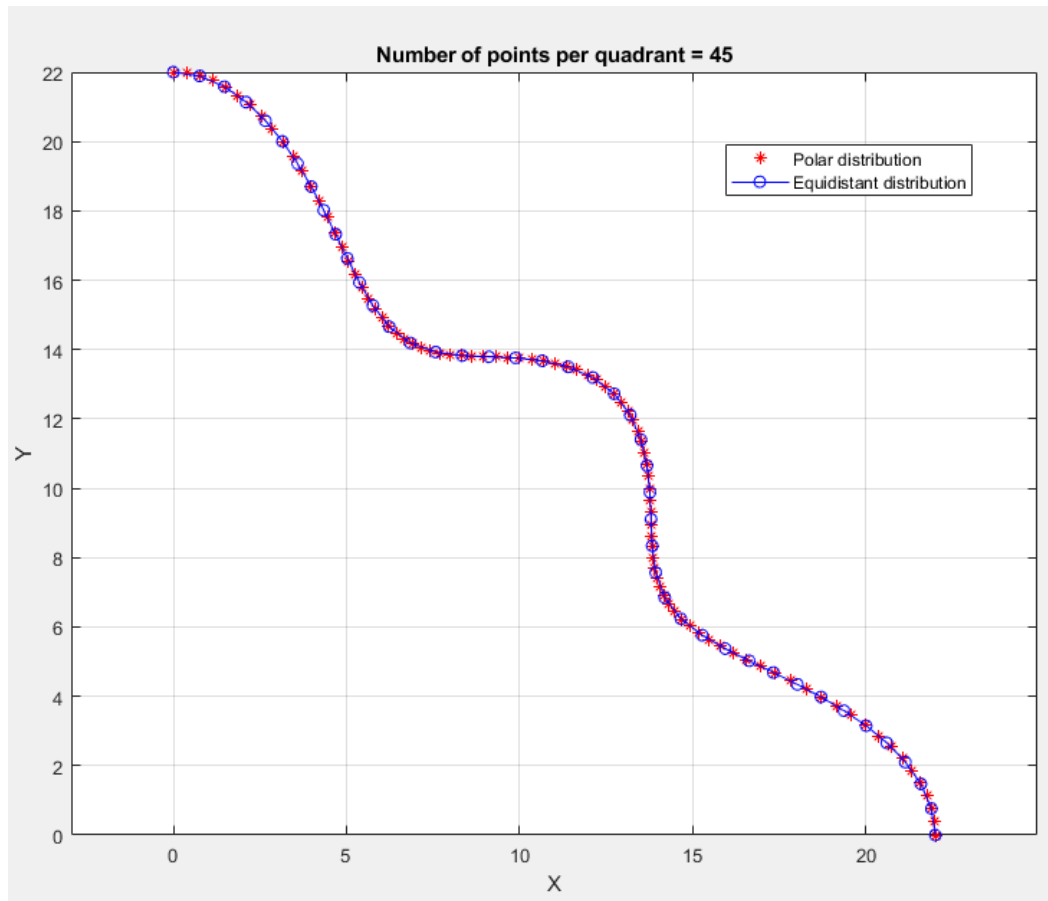


Figure 2.2.2.1 Comparison of equidistant and polar distributions of knots in the first quadrant for 45 knots per quadrant.

Distance between knots was also calculated, but given the method of generating equidistant points the distance between the points could not be varied directly. When generating knots the number of knots per one quadrant was decisive for distance between knots. Number of knot per quadrant were chosen to progressively decrease. The arc length of the curve was calculated previously therefore it was also possible to calculate number of points per length. After milling it will enable us to evaluate how this value influences the precision of the chosen interpolation method.

Table 2.2.2.1 Influence of number of points on distance between the knots for equidistant distribution

number of knots per quadrant	90	45	15
number of knots per contour	357	177	57
distance between knots (mm)	0,38148	0,76943	2,38929

Three variations of equidistant distribution are represented as follow:

- 90 knots per quadrant are represented on Figure 2.2.2.2 and Figure 2.2.2.3,

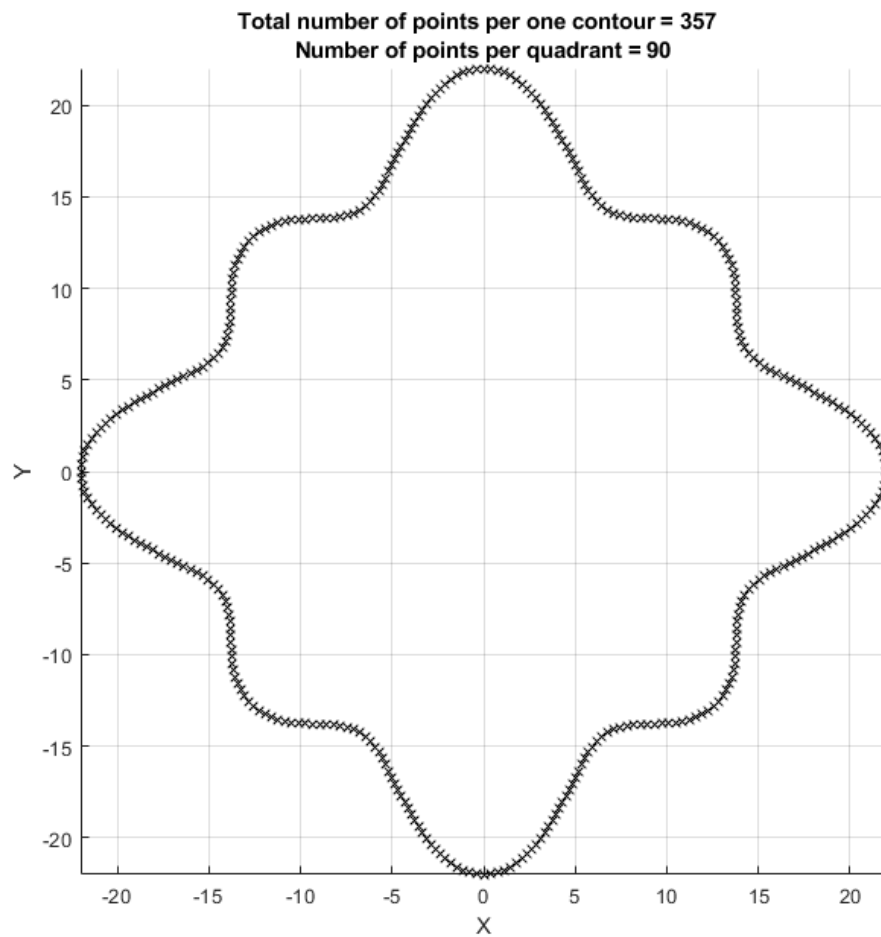


Figure 2.2.2.2 Equidistant distribution of knots for 90 knots per one quadrant.

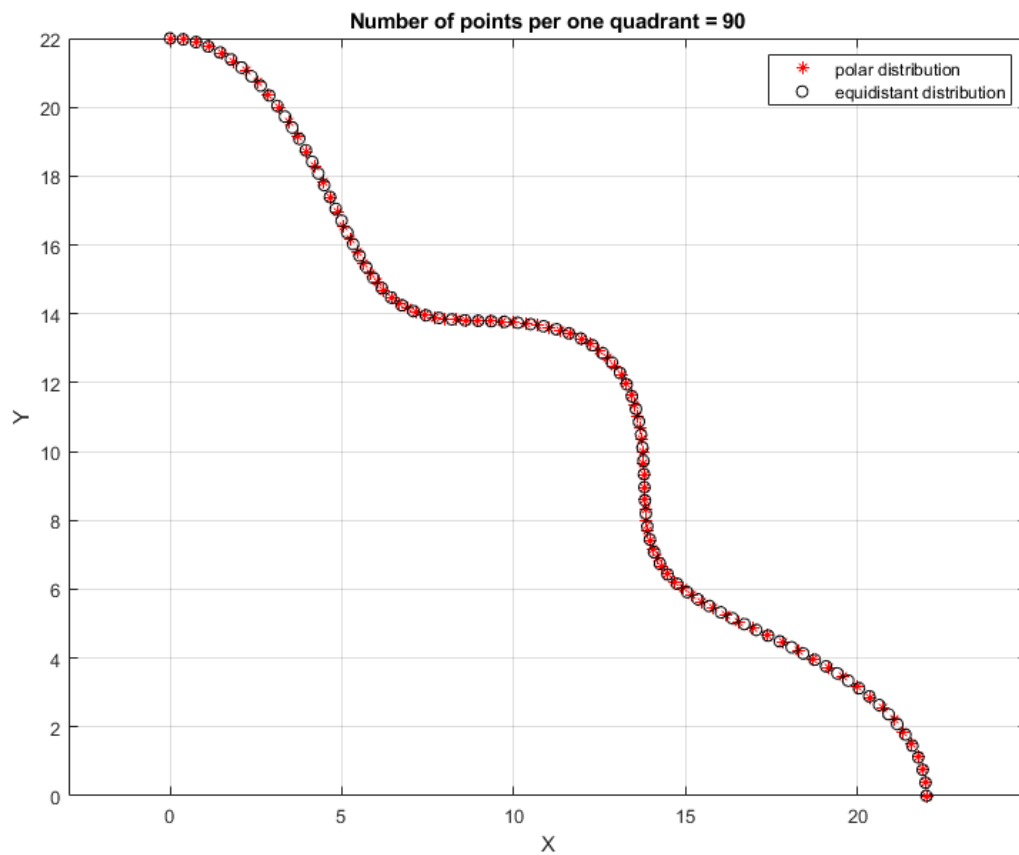


Figure 2.2.2.3 Equidistant distribution of knots represented in the first quadrant for 90 points per quadrant.

- 45 knots per quadrant are represented on Figure 2.2.2.4 and Figure 2.2.2.5,

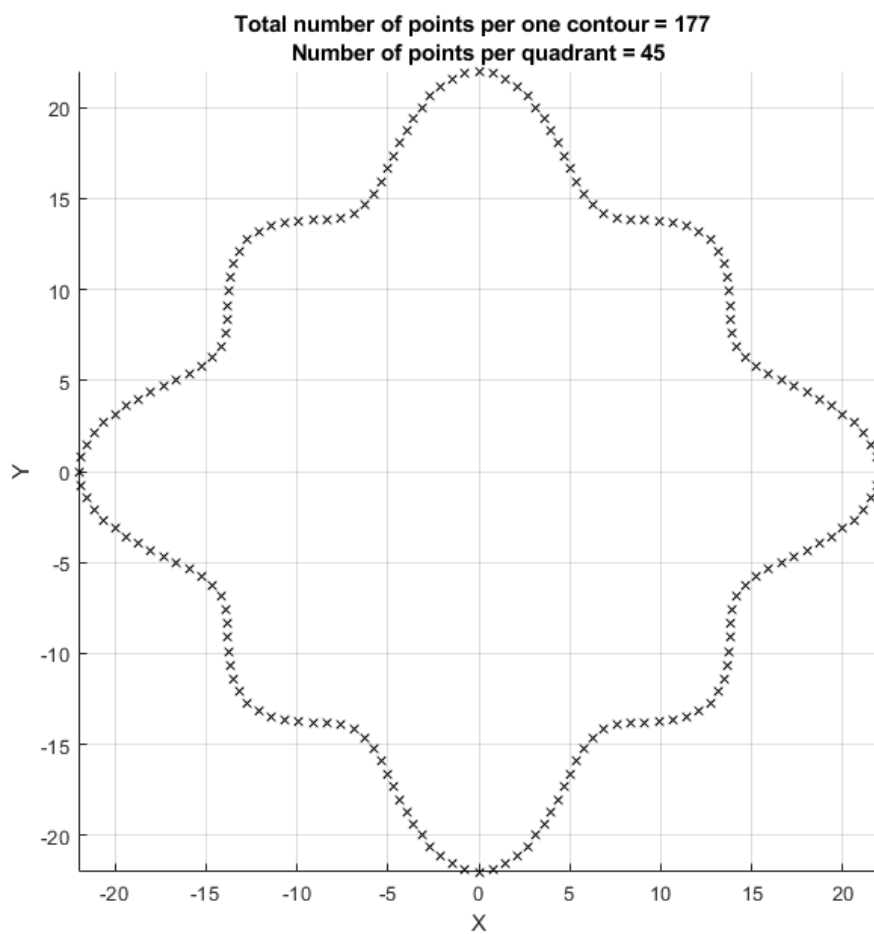


Figure 2.2.2.4 Equidistant distribution of knots for 45 knots per one quadrant.

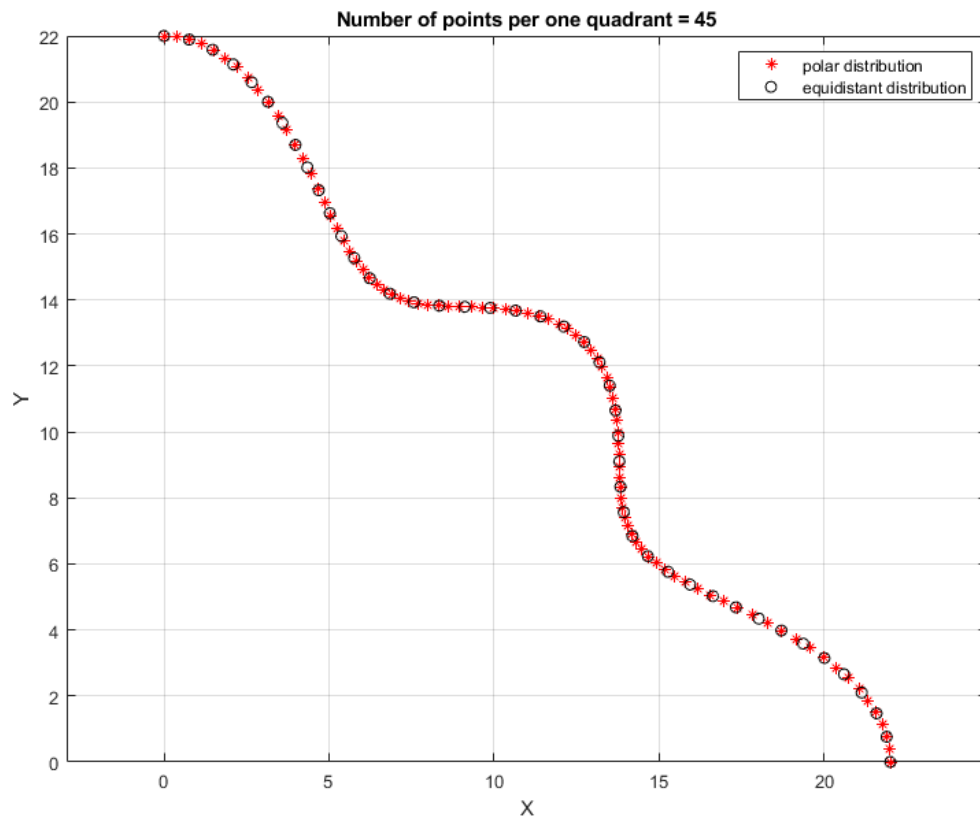


Figure 2.2.2.5 Equidistant distribution of knots represented in the first quadrant for 45 points per quadrant.



- 155 knots per quadrant are represented on Figure 2.2.2.6 and Figure 2.2.2.7.

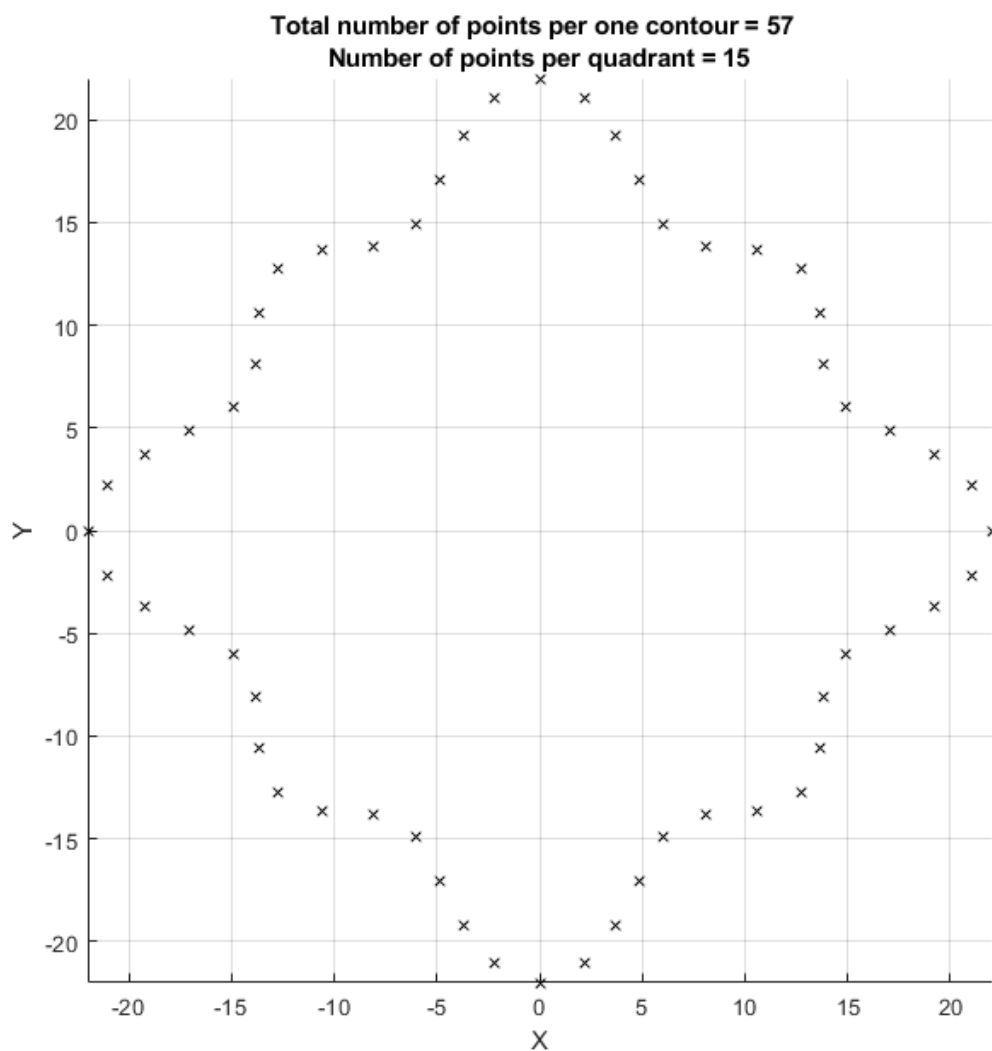


Figure 2.2.2.6 Equidistant distribution of knots for 15 knots per one quadrant.

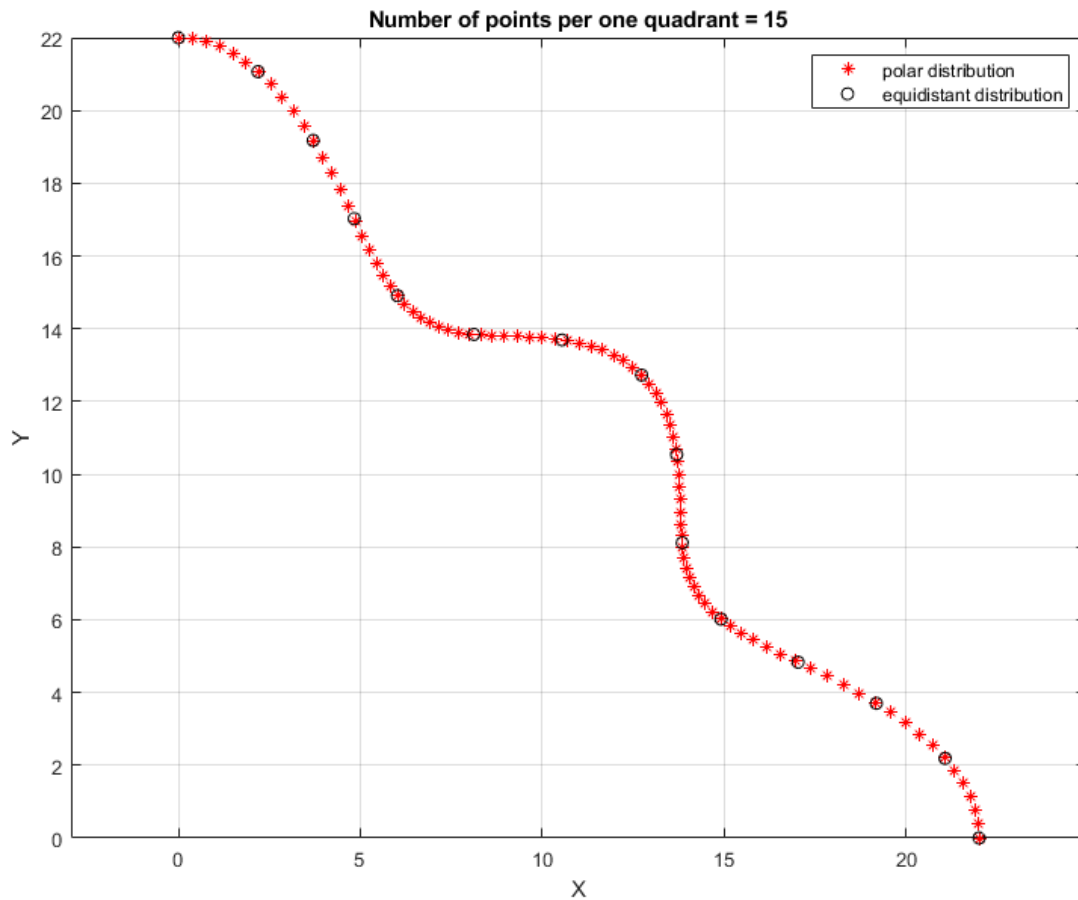


Figure 2.2.2.7 Equidistant distribution of knots represented in the first quadrant for 15 points per quadrant.

### 2.2.3 2D and 3D contouring

Two types of contouring were used: 2D and 3D contouring. Both methods aimed to machine the chosen contour to depth of -6 mm (in z axis) and employed the knots generated previously.

#### 2.2.3.1 2D contouring

Firstly a method of 2D contouring was applied, with goal to mill the chosen contour with constant value of z coordinate  $z = -6$  mm. 2D contouring concerned only initial experimentation and only polar distribution of knots was applied for 2D contouring as it was decided to realise further experimentation directly on 3D contouring.

During the verification, some problems near the end of the curve occurred. The milled curve was deformed in proximity the end and the beginning of the curve near  $\theta = 2\pi$ . Splines are influenced by knots that precede and that ensue. Additional two sequences of knots were added in order to eliminate the defect. One sequence was added before the main sequence of knots generated previously and second after the main sequence of knots. The goal was to overlap the points to eliminate the problem further developed in 3.7.2. Number

of points added was varied and their influence was studied. New total number of knots after adding the two sequences is stated in Table 2.2.3.1 for polar distribution and in Table 2.2.3.2 for equidistant distribution of knots.

Table 2.2.3.1 Total number of knots for polar distribution of knots after adding knots

Parameter = steptheta	1°	5°	15°
number of knot per curve	360	72	24
number beginning knots	10	3	2
number of ending knots	10	3	2

Table 2.2.3.2 Total number of knots and distance between them for equidistant distribution of knots after adding knots.

number of knots per quadrant	90	45	15
number of knots per contour	357	177	57
number beginning knots	12	6	2
number of ending knots	12	6	2
total number of points	381	189	61
distance between knots (mm)	0,38148	0,76943	2,38929

### 2.2.3.2 3D contouring

To eliminate imperfection of the curve beginning and ending, different approach needed to be elaborated. Multiple strategies were possible. Tangential entry of the tool was necessary not to intervene with the contour when approaching it. Creation of a contour in cylindrical coordinate system was chosen as a solution. The strategy of progressive milling of the contour was chosen to be employed. To ensure entry of the tool that would be tangential to the desired curved, original 2D curve in polar coordinate system was transformed into a 3D curve in cylindrical coordinate system by adding z coordinate to every knot generated.

The cylindrical coordinate system is similar to the polar coordinate system previously described. It is possible to declare that polar coordinate system is a cylindrical coordinate system with constant value of  $z = 0$ . [28]

Position of a point A is defined by the distance from the axis z of a correspondent Cartesian coordinate system  $r_A$ ; angle between  $r_A$  and axis x of correspondent Cartesian system and its coordinate in axis z, that is identical to the z axis of the correspondent Cartesian coordinate system. A representation of cylindrical coordinate system is outlined in Figure 2.2.3.1.

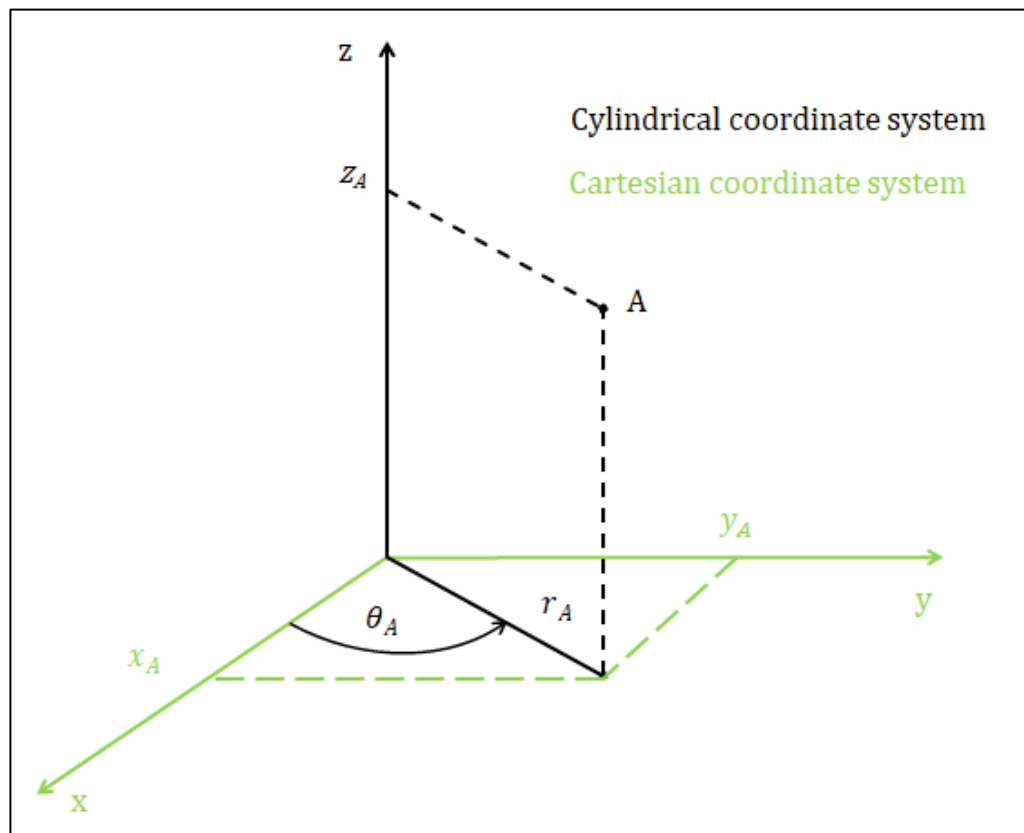


Figure 2.2.3.1 Cylindrical and Cartesian coordinate systems.

Points preceding the main contour were added at the beginning derived from the 3D contour with positive  $z$  coordinate were added for tangential entry of the tool.

The  $z$  coordinate of the main contour that gradually evolve from 0 mm to -6 mm were added. Then the tool circumscribe the contour with constant  $z$  coordinate equal to the desired depth,  $z = -6$  mm, before circumscribing the curve for the third time with gradually increasing value of  $z$  going from -6 mm to 0 mm. The tool encircles the contour in total of three times: first time for changing value of  $z$  form 0 to -6, then it encircles the contour one time for constant value of  $z$  to mill all the material and then it exits the contour by encircling the contour for the thirds time and without cutting. Points after the end of the third contour were also added to ensure tangential exit of the tool.

Total number of knots for steptheta equal to  $1^\circ$  is displayed on Figure 2.2.3.2 in 3D view and on Figure 2.2.3.4.

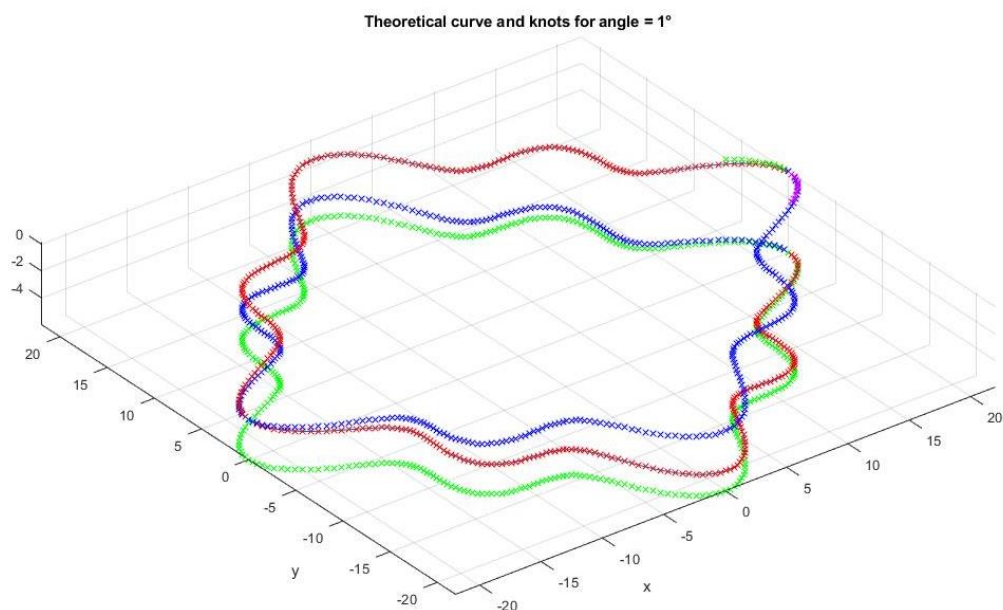


Figure 2.2.3.2 Integrity of knots for polar distribution method with a knot every 1°

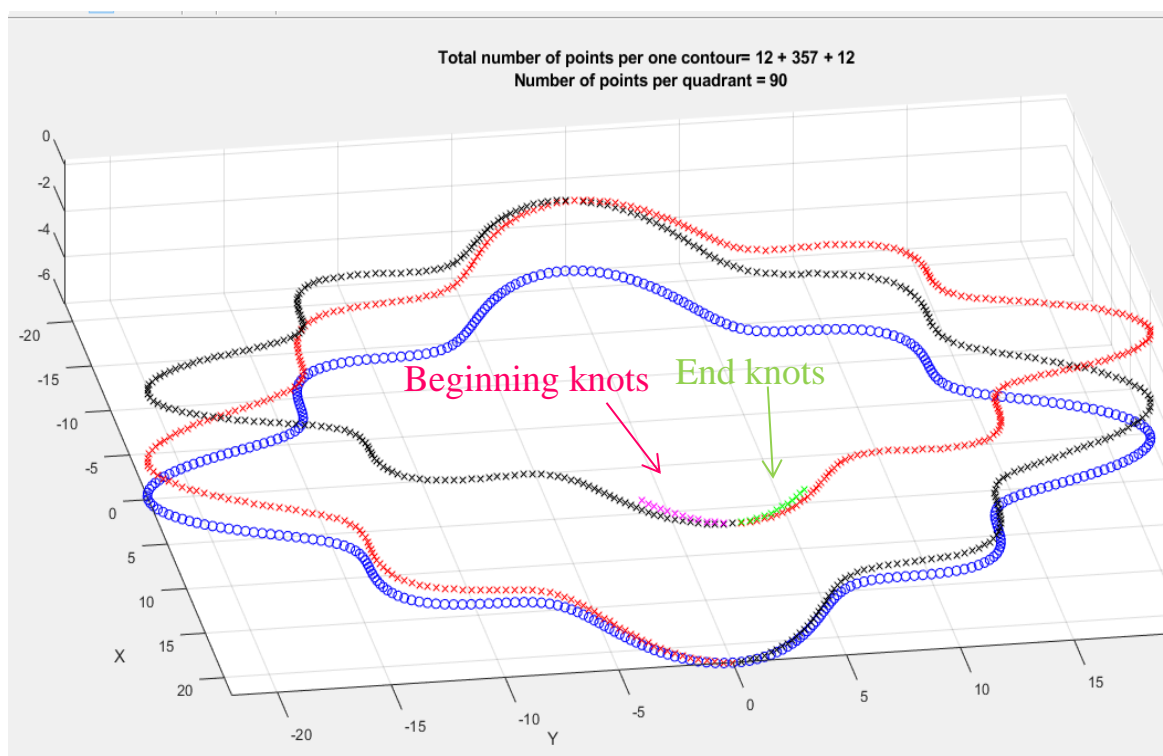


Figure 2.2.3.3 Additional knots for tangential entry of the tool.

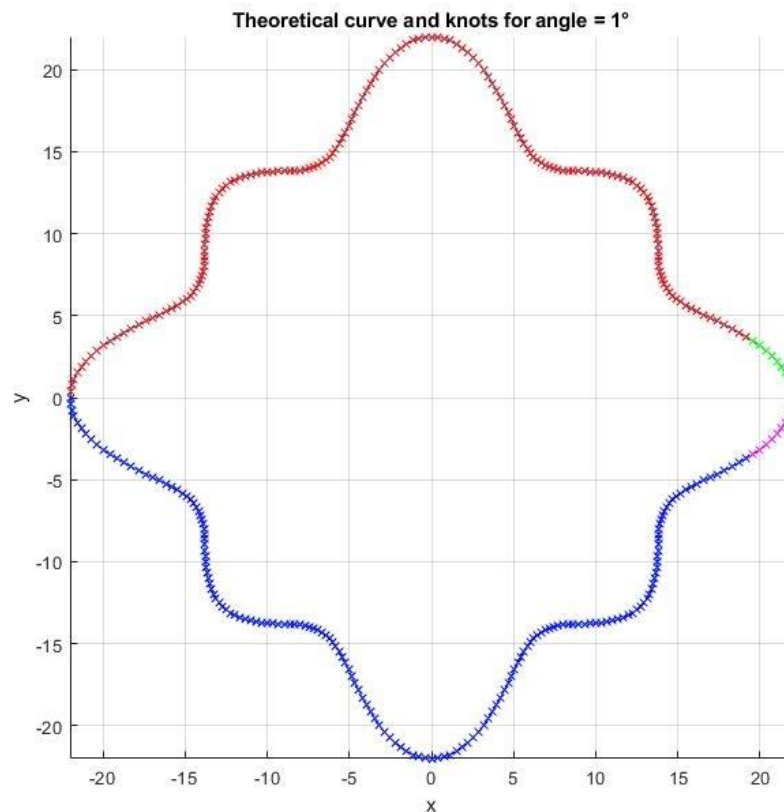


Figure 2.2.3.4 Integrity of points for polar distribution method with a knot every  $1^\circ$  viewed from above

For both distributions, the same method to transform 2D curve into a 3D curve was implemented. Vectors of x and y coordinates stayed the same and the vector of z coordinates was generated three times: firstly z gradually varies from 0 to -6, then one with constant value of  $z = -6.000$  mm and finally third vector where z varies from -6 mm to 0 mm.

For the polar distribution, same x and y coordinates as the 2D variation were used and only Z coordinates were added. The method was the same as for the 2D contour meaning a knot was generated every  $1^\circ$ ,  $5^\circ$  or every  $15^\circ$ .

The number of knots stated in the tables is always number of knots per one circumscription of the contour. This means that for one complete 3D milling the total number of points in the subprogram is equal to summation of knots at the beginning, knots at the end and three times the number of points per one contour.

### 2.3 Creation of programs

Types of programs were written to realise the machining: main program and corresponding subprograms. Using a subprogram aims to shorten the main program making it more understandable and overall less cluttered.

Main programs were written directly on the machines control station. The structure of all main programs is the same. The main programs contain all the necessary machine conditions. Then they call an interpolation method (linear or particular spline) and correspondent subprograms containing the knots in correct syntax adapted for the interpolation method.

All subprograms were generated in MATLAB and imported as .MFP files readable by the CNC machine. The subprograms containing only knots coordinates were called upon by the main programs after calling according interpolation method.

### 2.3.1 Methodology of creation of subprograms

Once the knots were generated, it was necessary to insert knots coordinates to a NC

For linear interpolation the command for linear interpolation function G1 was included in the subprograms.

For A-spline and C-spline the subprograms were identical, because the syntax for indicating its knots is same after calling the spline method in the main program.

For B-spline, a weight of knots was added. Constant weight  $PW = 3$  was added to all knots in B-spline subprogram. It's the maximal programmable weight meaning it draws the spline curve as close to the knots as possible. Add the weight is crucial as we want the milled curve to pass through the knots or at least be as close to the theoretical knots as possible.

The correct syntax is demonstrated in Table 2.3.1.1.

Table 2.3.1.1 Syntax for knots for different interpolation methods for Sinumerik control system.

Interpolation method	syntax
A-spline, C-spline	N... X... Y... Z-6.000
B-spline	N... X... Y... Z-6.000 PW=3
linear interpolation	G1 X... Y... Z...

### 2.3.2 Programs for 2D contouring

Every main program for 2D contouring firstly calls in total of seven subprograms of rough specimen containing knots on seven gradually smaller contours and then calls the finishing subprogram with knots situated of the final contour.

#### 2.3.2.1 Rough specimen for 2D contouring

For rough specimen other augmented curves were generated to mill material around the final curve to prepare the workpiece for milling of the studied contour without interfering with the results. Number of contours necessary for the rough specimen was calculated based on the diameter of the tool. When milling into aluminium alloy, it is important to respect that no more than 70% of the diameter of the tool should be entering the workpiece. Contours of rough specimen for 2D contouring are visualized in Figure 2.3.2.1.1.

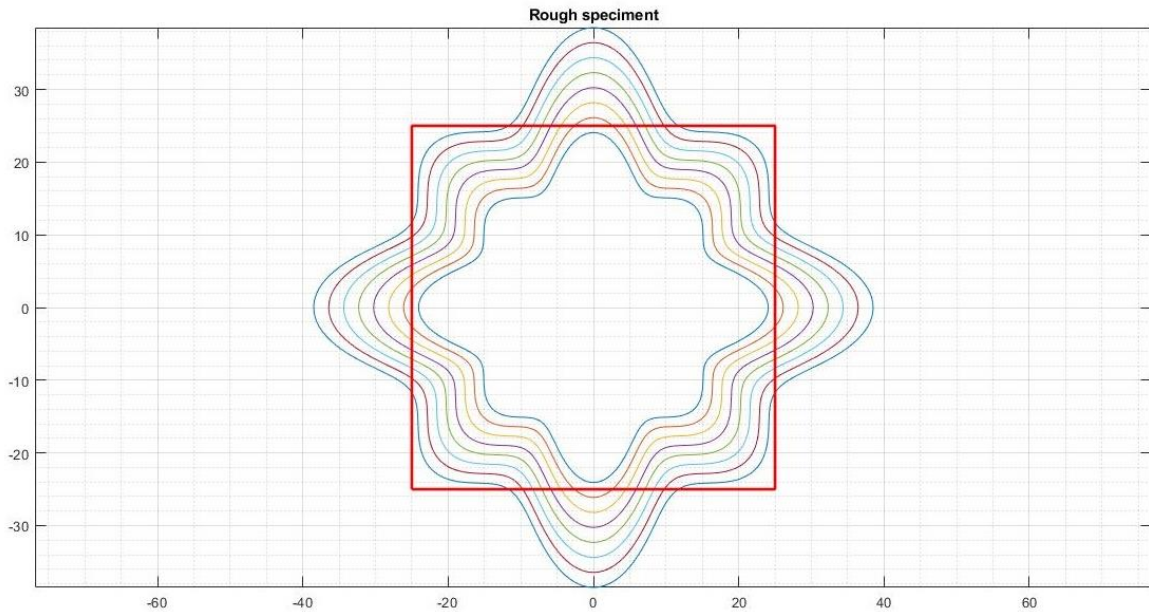


Figure 2.3.2.1.1 Rough specimen contours for 2D contouring.

Table 2.3.2.1.1 Examples of main programs for 2D contouring.

LINEAR	A-SPLINE
N10 G54	N10 G54
N20 T="FR_4"	N20 T="FR_4"
N30 M6	N30 M6
N40 G64	N40 G64
N50 SOFT	N50 SOFT
N60 M3 S17000 M8	N60 M3 S17000 M8
N70 G1 Z10 F5000	N70 G1 Z10 F5000
N80 G1 Z-2 F2000	N80 G1 Z-2 F2000
N90 G1 X40 Y0 F1270	N90 G1 X40 Y0 F1270
N100 G42	N100 G42
N110 G1 X37	N110 G1 X37
;N115 GOTOF LABEL_1	BTAN ETAN
N120 POLAR_STEPHTHETA_1_LIN_1	N120 ASPLINE
N130 POLAR_STEPHTHETA_1_LIN_2	;N130 GOTOF LABEL_1
N140 POLAR_STEPHTHETA_1_LIN_3	N140 POLAR_STEPHTHETA_15_1
N150 POLAR_STEPHTHETA_1_LIN_4	N150 POLAR_STEPHTHETA_15_2
N160 POLAR_STEPHTHETA_1_LIN_5	N160 POLAR_STEPHTHETA_15_3
N170 POLAR_STEPHTHETA_1_LIN_6	N170 POLAR_STEPHTHETA_15_4
N180 POLAR_STEPHTHETA_1_LIN_7	N190 POLAR_STEPHTHETA_15_5
;N185 LABEL_1:	N200 POLAR_STEPHTHETA_15_6
N190 G1 X22 Y0	N210 POLAR_STEPHTHETA_15_7
N200 POLAR_STEPHTHETA_1_LIN_8	;N220 LABEL_1:
N210 G40	N225 G1 X22 Y2 Z10
N220 G1 X40 Y0 F700	N230 G1 X17.387 Y-4.659 Z0.250
N230 G1 Z10 F5000	G42
N240 M5	BTAN ETAN
N250 M30	N235 ASPLINE
	N240 POLAR_STEPHTHETA_15_8
	N250 G40



	N260 G1 X40 Y0 F700 N270 G1 Z10 F5000 N280 M5 N290 M30
<b>B-SPLINE</b>	<b>C-SPLINE</b>
N10 G54 N20 T="FR_4" N30 M6 N40 G64 N50 SOFT N60 M3 S17000 M8 N70 G1 Z10 F5000 N80 G1 Z-2 F2000 N90 G1 X40 Y0 F1270 N100 G42 N110 G1 X37 ;N115 GOTOF LABEL_1 n120 BSPLINE N125 POLAR_STEPHTHETA_1_B_1 N130 POLAR_STEPHTHETA_1_B_2 N140 POLAR_STEPHTHETA_1_B_3 N150 POLAR_STEPHTHETA_1_B_4 N160 POLAR_STEPHTHETA_1_B_5 N170 POLAR_STEPHTHETA_1_B_6 N180 POLAR_STEPHTHETA_1_B_7 ;N185 LABEL_1: N190 G1 X22 Y0 N200 BSPLINE N205 POLAR_STEPHTHETA_1_B_8 N210 G40 N220 G1 X40 Y0 F700 N230 G1 Z10 F5000 N240 M5 N250 M30	N10 G54 N20 T="FR_4" N30 M6 N40 G64 N50 SOFT N60 M3 S17000 M8 N70 G1 Z10 F5000 N80 G1 Z-2 F2000 N90 G1 X40 Y0 F1270 N100 G42 N110 G1 X37 BTAN ETAN N120 ASPLINE ;N130 GOTOF LABEL_1 N140 POLAR_STEPHTHETA_15_1 N150 POLAR_STEPHTHETA_15_2 N160 POLAR_STEPHTHETA_15_3 N170 POLAR_STEPHTHETA_15_4 N190 POLAR_STEPHTHETA_15_5 N200 POLAR_STEPHTHETA_15_6 N210 POLAR_STEPHTHETA_15_7 ;N220 LABEL_1: N225 G1 X22 Y2 Z10 N230 G1 X17.387 Y-4.659 Z0.250 G42 BTAN ETAN N235 ASPLINE N240 POLAR_STEPHTHETA_15_8 N250 G40 N260 G1 X40 Y0 F700 N270 G1 Z10 F5000 N280 M5 N290 M30

### 2.3.2.2 Finishing for 2D contouring

Finishing consisted of one circumscription of the eight and smallest contour including the overlap knots at the beginning and the end.

Extract from subprograms for finishing for 2D contouring are stated in Table 2.3.2.2.

Table 2.3.2.2 Extracts from subprograms of finishing for 2D contouring for 4 studied interpolation method and stepheta 5° with total of 72 knots.

Syntax of subprograms for A-spline and C-spline	Syntax of subprograms for B-spline
N10 X22.000 Y0.000 Z-6.000	N10 X22.000 Y0.000 Z-6.000 PW=3
N20 X21.330 Y1.866 Z-6.000	N20 X21.330 Y1.866 Z-6.000 PW=3
N30 X19.577 Y3.452 Z-6.000	N30 X19.577 Y3.452 Z-6.000 PW=3
N40 X17.387 Y4.659 Z-6.000	N40 X17.387 Y4.659 Z-6.000 PW=3
N50 X15.475 Y5.632 Z-6.000	N50 X15.475 Y5.632 Z-6.000 PW=3
N60 X14.295 Y6.666 Z-6.000	N60 X14.295 Y6.666 Z-6.000 PW=3
N70 X13.856 Y8.000 Z-6.000	N70 X13.856 Y8.000 Z-6.000 PW=3
N80 X13.774 Y9.645 Z-6.000	N80 X13.774 Y9.645 Z-6.000 PW=3
N90 X13.523 Y11.347 Z-6.000	N90 X13.523 Y11.347 Z-6.000 PW=3
N100 X12.728 Y12.728 Z-6.000	N100 X12.728 Y12.728 Z-6.000 PW=3
...	...
Syntax of subprograms for linear interpolation	
N10 G1 X22.000 Y0.000 Z-6.000	
N20 G1 X21.330 Y1.866 Z-6.000	
N30 G1 X19.577 Y3.452 Z-6.000	
N40 G1 X17.387 Y4.659 Z-6.000	
N50 G1 X15.475 Y5.632 Z-6.000	
N60 G1 X14.295 Y6.666 Z-6.000	
N70 G1 X13.856 Y8.000 Z-6.000	
N80 G1 X13.774 Y9.645 Z-6.000	
N90 G1 X13.523 Y11.347 Z-6.000	
N100 G1 X12.728 Y12.728 Z-6.000	
...	

### 2.3.3 Programs for 3D contouring

For both polar and equidistant distribution, X and Y coordinated are the same as for the 2D solution, only Z coordinate was added accordingly as described in 2.2.3.2.

For the 3D contour, rough specimen and final contour were called by different main programs. Rough specimen main programs called 4 subprograms with correspondent method for 2D contours with constant z coordinate  $z = -6.000$  mm. Final contour main programs called only the corresponding last contour subprogram with the 3D contour.

#### 2.3.3.1 Rough specimen for 3D solution

For rough specimen to prepare the workpiece, 2D contours were applied with Z coordinate constant and equal to  $Z = -6.000$  mm. Value of steptheta for rough specimen was chosen to have total number of knots per curve similar to later applied finishing method. For example for finishing with equidistant distribution with 177 points, rough specimen with steptheta  $2^\circ$  was employed, with 180 points per contour was applied. Interpolation method in rough specimen corresponded to interpolation method in finishing.

Because the workpieces were made form artificial wood, it was possible to realise rough specimen with only 4 contours as it was possible not to respect the rule of 70% of the diameter of the tool machining.

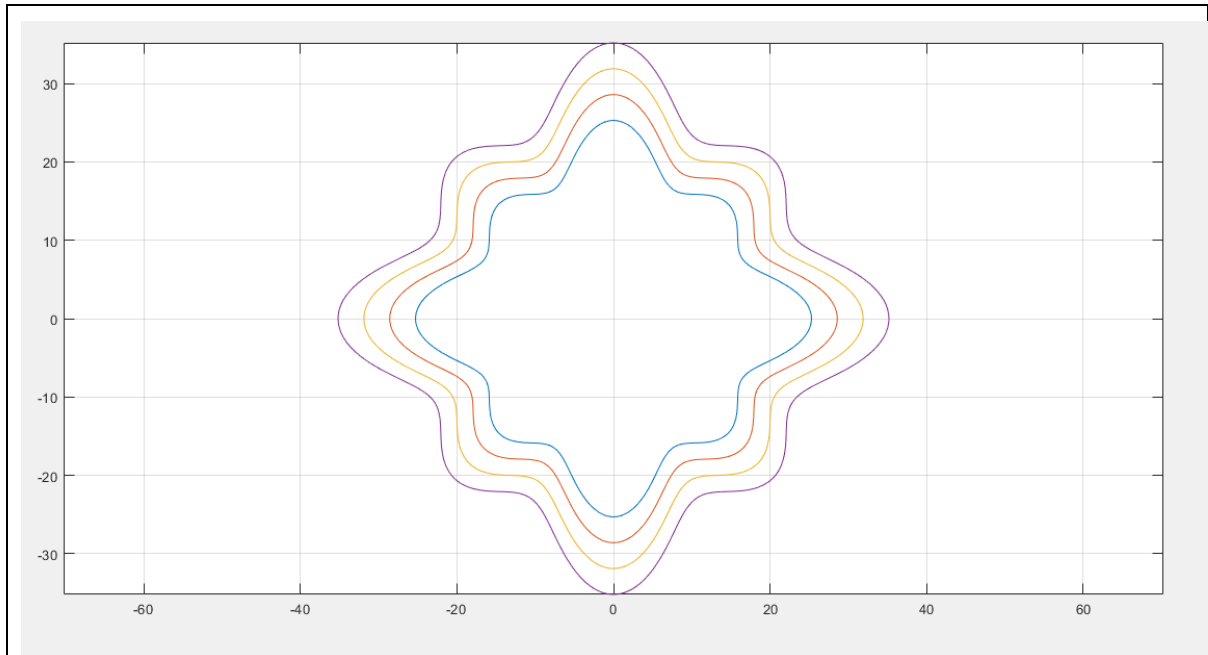


Figure 2.3.3.1 Curves for rough specimen for 3D contouring.

Correspondent 2D rough specimen programs employed before 3D finishing programs are stated in Table 2.3.3.1.

Table 2.3.3.1 Rough specimens used before finishing 3D contouring.

3D finishing method	Rough specimen method
Equidistant, 357 knots, Linear	2D Polar, 360 knots, Linear
Equidistant, 357 knots, A-spline	2D Polar, 360 knots, A-spline
Equidistant, 357 knots, B-spline	2D Polar, 360 knots B-spline
Equidistant, 357 knots, C-spline	2D Polar, 360 knots, C-spline
Equidistant, 177 knots, Linear	2D Polar, 180 knots, Linear
Equidistant, 177 knots, A-spline	2D Polar, 180 knots, A-spline
Equidistant, 177 knots, B-spline	2D Polar, 180 knots, B-spline
Equidistant, 177 knots, C-spline	2D Polar, 180 knots C-spline
Equidistant, 57 knots, Linear	2D Polar, 72 knots, Linear
Equidistant, 57 knots, A-spline	2D Polar, 72 knots, A-spline
Equidistant, 57 knots, B-spline	2D Polar, 72 knots, B-spline
Equidistant, 57 knots, C-spline	2D Polar, 72 knots 5°, C-spline

Examples of rough specimen main programs are listed in Table 2.3.3.2.

Table 2.3.3.2 Rough specimen main programs.

Rough specimen main program, linear interpolation, polar distribution, stepheta 5°	Rough specimen main program, A-spline interpolation, polar distribution, stepheta 5°
N10 G54 N20 T="FR_4" N30 M6 N40 G64 N50 SOFT N60 M3 S17000 M8 N70 G1 Z10 F5000 N80 G1 X40 Y0 F1530 N90 G42 N100 G1 X37 N110 HRUB_1_LIN_1 N120 HRUB_1_LIN_2 N130 HRUB_1_LIN_3 N140 HRUB_1_LIN_4 N150 G40 N160 G1 X40 Y0 F700 N170 G1 Z10 F5000 N180 M5 N190 M30	N10 G54 N20 T="FR_4" N30 M6 N40 G64 N50 SOFT N60 M3 S17000 M8 N70 G1 Z10 F5000 N80 G1 X40 Y40 F1530 N90 G42 N100 G1 X37 N110 BTAN ETAN N120 ASPLINE N130 HRUB_1_1 N140 HRUB_1_2 N150 HRUB_1_3 N160 HRUB_1_4 N170 G40 N180 G1 X40 Y0 F700 N190 G1 Z10 F5000 N200 M5 N210 M30
Rough specimen main program, B-spline, polar distribution, stepheta 5°	Rough specimen main program, C-spline, polar distribution, stepheta 5°
N10 G54 N20 T="FR_4" N30 M6 N40 G64 N50 SOFT N60 M3 S17000 M8 N70 G1 Z10 F5000 N80 G1 X40 Y0 F1530 N90 G42 N100 G1 X37 N110 BSPLINE N120 HRUB_1_B_1 N130 HRUB_1_B_2 N140 HRUB_1_B_3 N150 HRUB_1_B_4 N160 G40 N170 G1 X40 Y0 F700 N180 G1 Z10 F5000 N190 M5 N200 M30	N10 G54 N20 T="FR_4" N30 M6 N40 G64 N50 SOFT N60 M3 S17000 M8 N70 G1 Z10 F5000 N80 G1 X40 Y0 F1530 N90 G42 N100 G1 X37 N110 BTAN ETAN N120 CSPLINE N130 HRUB_1_1 N140 HRUB_1_2 N150 HRUB_1_3 N160 HRUB_1_4 N170 G40 N180 G1 X40 Y0 F700 N190 G1 Z10 F5000 N200 M5 N210 M30

### 2.3.3.2 Finishing for 3D contouring

Finishing for 3D contouring used its own main programs each containing calling correspondent subprogram with knots with non-constant value of Z as described in 2.2.3.2.

Table 2.3.3.3 Example of finishing programs: 3D contouring, equidistant distribution with parameter 45.

Main program for linear interpolation	Main program for A-spline
<pre> N10 G54 N20 T="FR_4" N30 M6 N40 G64 N50 SOFT N60 M3 S17000 M8 N70 G1 Z10 F1530 N80 M0 N90 G4 F2 N100 G42 N110 G1 X37 Y-15 N120 G1 X13.856 Y-8.000 Z0.500 N130 EKVI_POSTUPNE_LIN_45 N140 G40 N150 G1 X30 Y30 F700 N160 G1 Z10 F5000 N170 Y100 N180 M5 N190 M30 </pre>	<pre> N10 G54 N20 T="FR_4" N30 M6 N40 G64 N50 SOFT N60 M3 S17000 M8 N70 G1 Z10 F1530 N80 M0 N90 G4 F2 N100 G42 N110 G1 X37 N120 BTAN ETAN N130 ASPLINE N140 X13.856 Y-8.000 Z0.500 N150 EKVI_POSTUPNE_45 N160 G40 N170 G1 X30 Y30 F700 N180 G1 Z10 F5000 N190 Y100 N200 M5 N210 M30 </pre>
Main program for B-spline	Main program for C-spline
<pre> N10 G54 N20 T="FR_4" N30 M6 N40 G64 N50 SOFT N60 M3 S17000 M8 N70 G1 Z10 F1530 N80 M0 N90 G4 F2 N100 G42 N110 G1 X37 N120 BSPLINE N130 X13.856 Y-8.000 Z0.500 PW=3 N140 EKVI_POSTUPNE_B_45 N150 G40 N160 G1 X30 Y30 F700 N170 G1 Z10 F5000 N180 Y100 N190 M5 N200 M30 </pre>	<pre> N10 G54 N20 T="FR_4" N30 M6 N40 G64 N50 SOFT N60 M3 S17000 M8 N70 G1 Z10 F1530 N80 M0 N90 G4 F2 N100 G42 N110 G1 X40 N115 G1 X37 N120 BTAN ETAN N130 CSPLINE N140 X13.856 Y-8.000 Z0.500 N150 EKVI_POSTUPNE_45 N160 G40 N170 G1 X30 Y30 F700 N180 G1 Z10 F5000 N190 Y100 N200 M5 N210 M30 </pre>

### 3 EXPERIMENTAL VERIFICATION OF THE CNC PROGRAM WITH SPLINE APPLICATION

#### 3.1 CNC machine

The CNC machine used to implement spline interpolation was TAJMAC-ZPS MCV 1210 with Sinumerik control system as displayed on Figure 2.3.3.1. It is a five axes milling machine with linear axes X, Y, Z and two rotational axes A and C (as described in 1.1.4). Basic characteristics of the machine are stated in Table 3.1.1. It's also equipped with an automatic tool changer that can contain up to 30 tools and touch probes RENISHAW OMP 400.

Table 3.1.1 Basic characteristics of TAJMAC-ZPS MCV 1210

	axis X	axis Y	axis Z	axis A	axis C
travel	1,000 mm	800 mm	600 mm	±115°	±200°
maximal velocity	40 m/min	40 m/min	40 m/min	60°/min	60°/min
maximal acceleration	5 m/s <sup>2</sup>	5 m/s <sup>2</sup>	5 m/s <sup>2</sup>		



Figure 2.3.3.1 TAJMAC-ZPS MCV 1210 milling machine.

## 3.2 Workpiece

### 3.2.1 Workpiece for 2D contouring

Firstly an aluminium alloy 7475-T7351 was chosen in order to be able to machine the contour more rapidly than if a steel alloy was used. Machining of an aluminium alloy also generally results in relatively better surface quality in shorter machining time. [19]

The dimensions of the workpiece were chosen to fit the curve and minimize the material used. A parallelepiped of following dimensions was used as a raw workpiece for 2D contouring: 45x45x25 mm.



Figure 3.2.1.1 Workpieces for 2D contouring.

### 3.2.2 Workpiece for 3D contouring

For further 3D contouring 24 cubes of dimensions 45x45x45 mm from artificial wood were prepared.



Figure 3.2.2.1 Workpieces for 3D contouring.

### 3.3 Selection of the tool

The radius of the tool could interfere with the desired machined surface if it was too large even though a function that compensates the radius is implemented in the program.

Maximal diameter of the tool was determined by calculating minimal radius curvature of the chosen curve which was previously found out to be equal to  $r_{\min} = 2.2966$  mm in 2.1.4. Therefore a tool with rayon equal to 2 mm was chosen.

The milling cutter employed in all experimentations was 04E3S50-12A04 SUMA manufactured by PRAMET displayed on Figure 3.3.1. In all NC programs it is referred to as “FR=4” because it was the indication entered in the NC system of the machine that identifies all tools in the ATC.



Figure 3.3.1 Milling cutter 04E3S50-12A04 SUMA by PRAMET [29]

### 3.4 Selection of cutting conditions

Cutting conditions later implemented in the program were set as follow in Table 3.4.1.

Table 3.4.1 Cutting conditions for both rough specimen and finishing.

N	machine speed in revolutions/minute (RPM)	17,000 1/min
$f_z$	feed/dent of cutter	0.025 mm/dent
z	Number of teeth of the tool	3
D	Diameter of the tool	4 mm
$v_f$	Feed	$v_f = 1,270$ mm/min
$v_c$	Cutting velocity	$v_c = 213.628$ m/min

Lubricant was used only for 2D contouring (machining of the aluminium alloy) to cool down the material and the tool in course of machining. As for 3D contouring, artificial wood was used as material of the workpiece and did not need lubricating or cooling in set conditions.

### 3.5 Procedure of the execution

For every method's execution, the same procedure was followed:

- simulation of the rough specimen program,
- simulation of the main program,
- clamping of the workpiece,



- rough specimen program execution,
- finishing program execution,
- unclamping of the workpiece,
- evaluation of machined surface positions under microscope.

### 3.6 Program simulation

All programs and subprograms were tested in simulation mode before actually machining them. Example of simulation is displayed on Figure 3.6.1, in this case it was only the final contour that was simulated, that's why there is still material visible around the contour.



Figure 3.6.1 Example of final contour program simulation

## 3.7 Experimental verification of spline interpolation on 2D contouring

### 3.7.1 Clamping and coordinate system

Clamping of the workpiece corresponded to the chosen coordinate system.

Setting of workpiece zero point was realised thanks to a built in touch probe RENISHAW OMP 400 and it was set to the centre of the face of the workpiece.

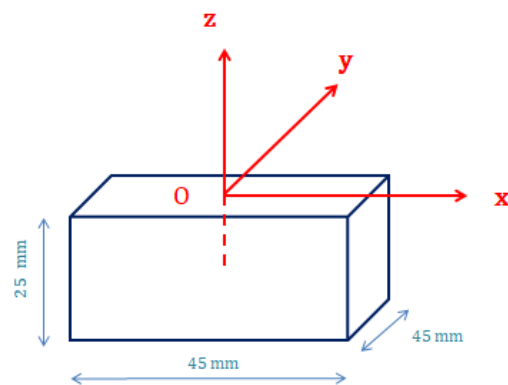


Figure 3.7.1.1 Coordinate system in relation to the workpiece for 2D contouring.

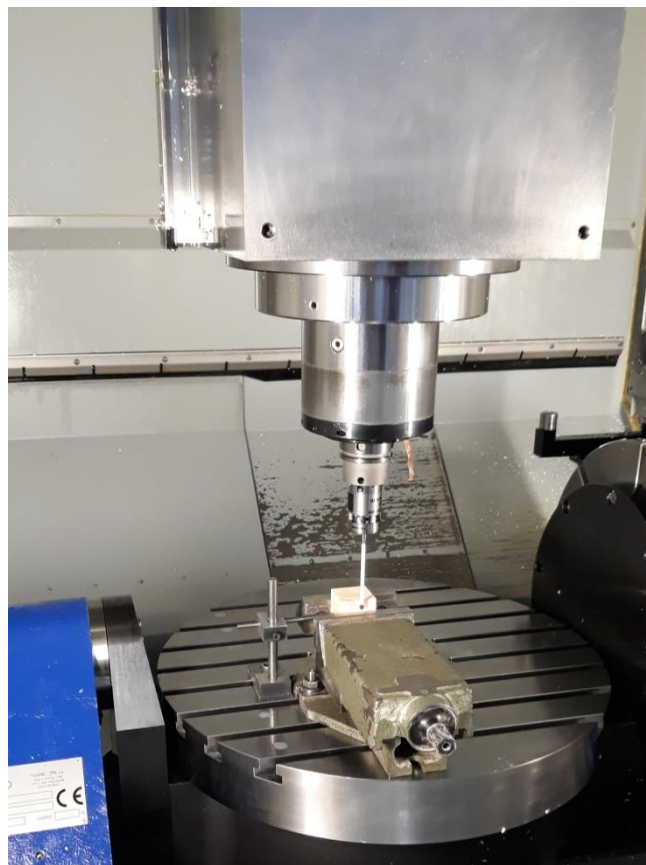


Figure 3.7.1.2 Setting the workpiece zero point with a built in probe RENISHAW.

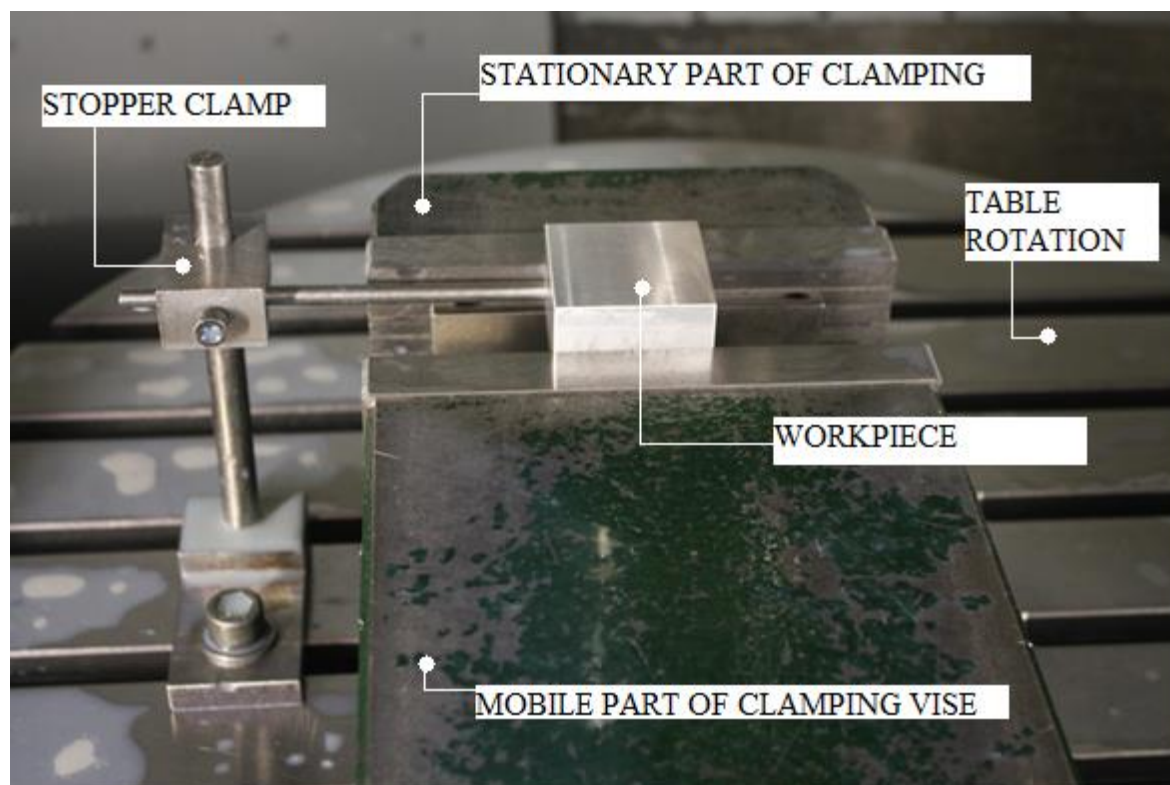


Figure 3.7.1.3 Clamping of the workpiece

### 3.7.2 Rough specimen execution of 2D contour

Firstly a rough specimen program was called to machine material surrounding the studied contour. It was important to ensure that rough specimen and the machining of the contour were realised in one clamping to maintain the same coordinate system of the workpiece set previously and not to move its origin. For every method a special rough specimen program was employed using the same method and the same density of knots as the finishing program, which eliminates interferences of the rough specimen curves with the studied curve. Examples of workpieces just after rough specimen milling are displayed on Figure 3.7.2.1, Figure 3.7.2.2 and Figure 3.7.2.3.

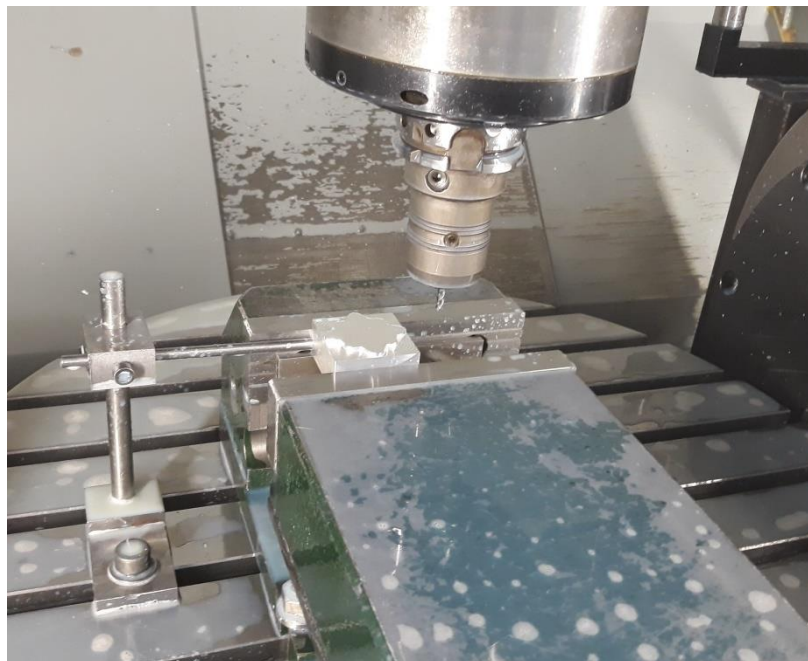


Figure 3.7.2.1 Workpiece after rough specimen.

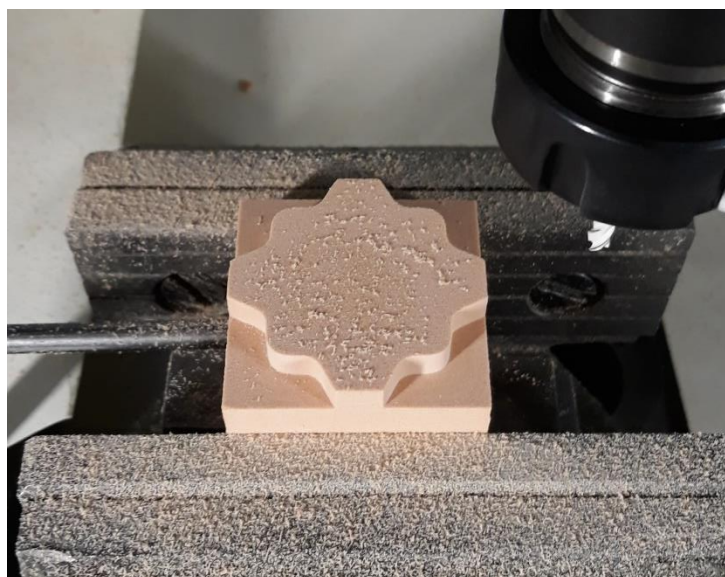


Figure 3.7.2.2 Workpiece from artificial wood after rough specimen, first view.

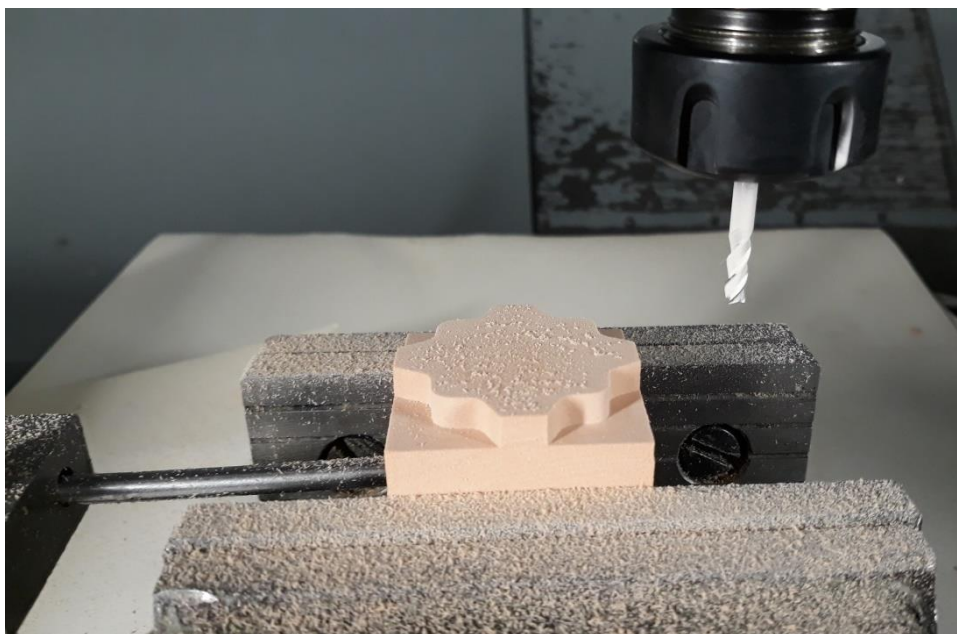


Figure 3.7.2.3 Workpiece from artificial wood after rough specimen, second view.

### 3.7.3 Finishing execution of 2D contour

After rough specimen, correspondent finishing program was executed without unclamping the workpiece. The machining time of finishing was measured.

### 3.8 Problems with 2D contours

Problems with connecting the beginning and the end of the contour posed a problem during the execution of the programs. The defect was more prominent with decreasing number of points.

Thanks to the simulations solutions to eliminate the defect at the end of the contour were tested without wasting material.

Multiple strategies with additional knots were employed in both simulation and on wooden workpieces in attempt to resolve the problem, but especially B-spline and A-spline did not manage to create a completely symmetrical contour. Additional knots are represented on Figure 3.8.1.

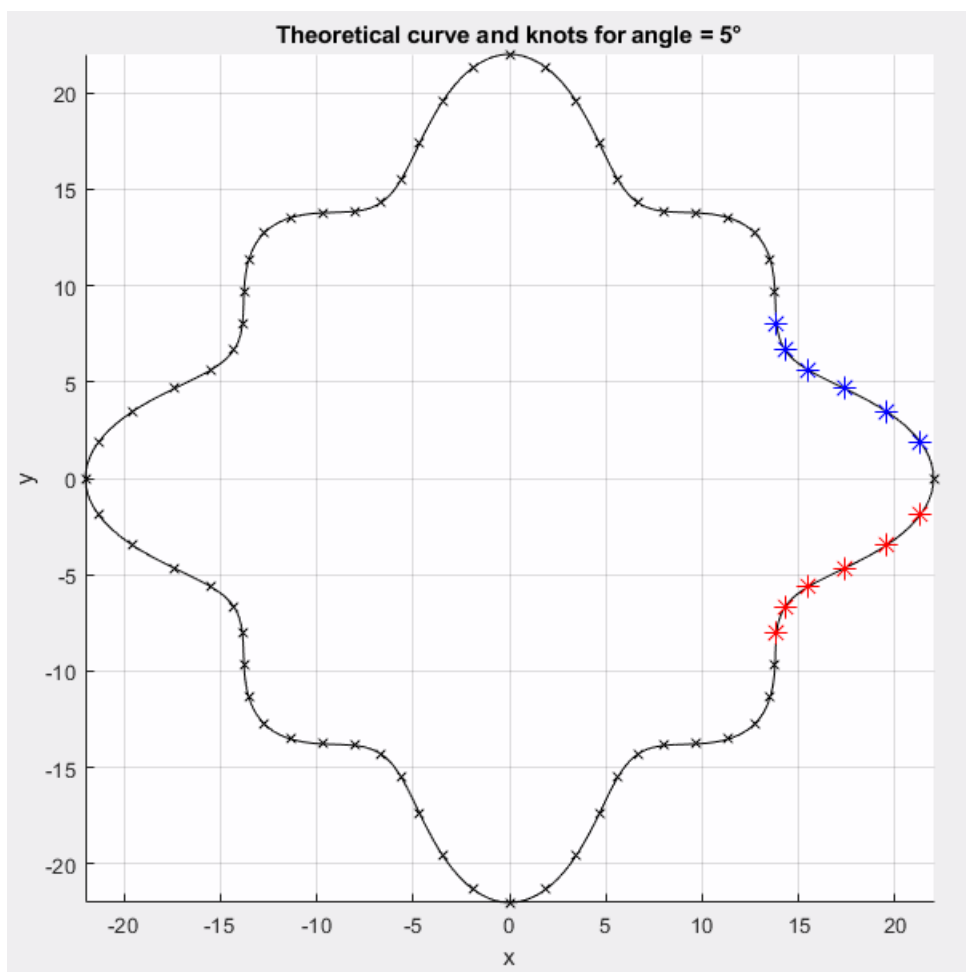


Figure 3.8.1 Adding overlap knots at the beginning (red) and at the end (blue).

Adding overlap knots and varying the number of overlap knots did ameliorate the problem, but did not eliminate it as seen on Figure 3.8.2. and Figure 3.7.3.3.

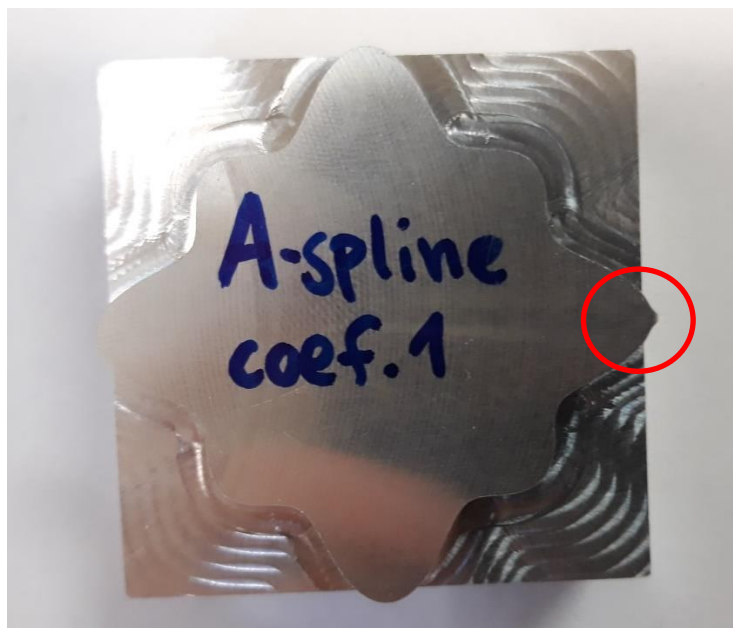


Figure 3.8.2 Problematic area for 2D contouring.

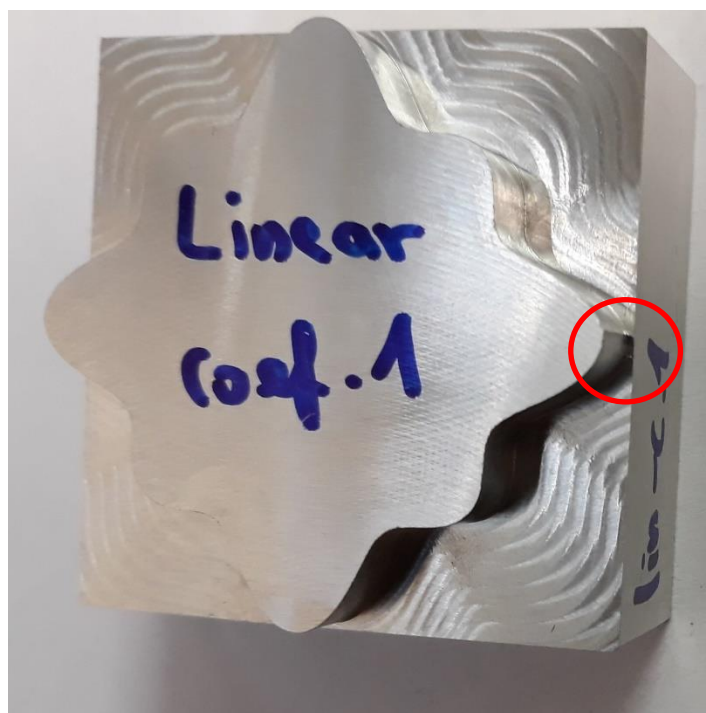


Figure 3.7.3.3 Defect at the end of the curve for linear interpolation.

First strategy was to add points after the end of contour repeating the same knots as at the beginning of the contour. This method was found to be semi-efficient, but the contour was still deformed at the end area. The solution of 3D contouring presented previously followed.

### 3.9 Using 3D contouring as a solution

Progressive milling described in 2.2.3.2 was chosen to solve the problems of the tool exiting the workpiece without interfering with the desired curve form. 24 wooden workpieces of cubic form were used. The workpiece zero point was changed accordingly as visualised on Figure 3.9.1.

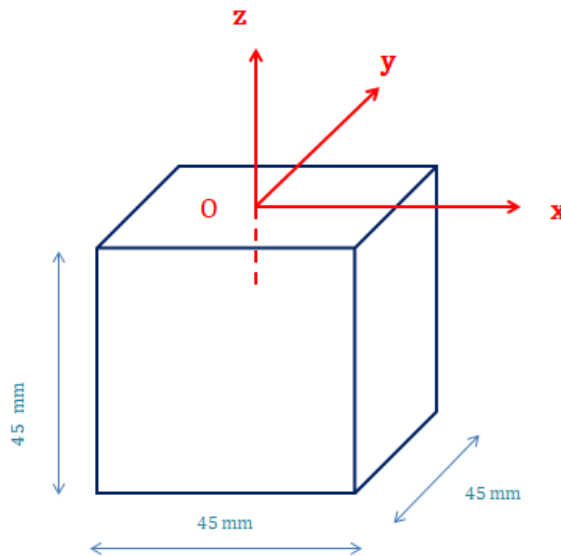


Figure 3.9.1 Representation of the Cartesian coordinate system in relation to the workpiece for 3D solution

The programs for 3D contouring were then employed. Four workpieces using four different interpolation method (A-spline, B-spline, C-spline and linear interpolation) were compared for every variation of knots. In total of 12 workpieces for polar distribution and 12 workpieces for equidistant distribution were obtained.

In course of milling, the actual value of feed rate was observed and machining time of the finishing program execution was measured.

24 machined contours were obtained as is displayed on Figure 3.9.2.



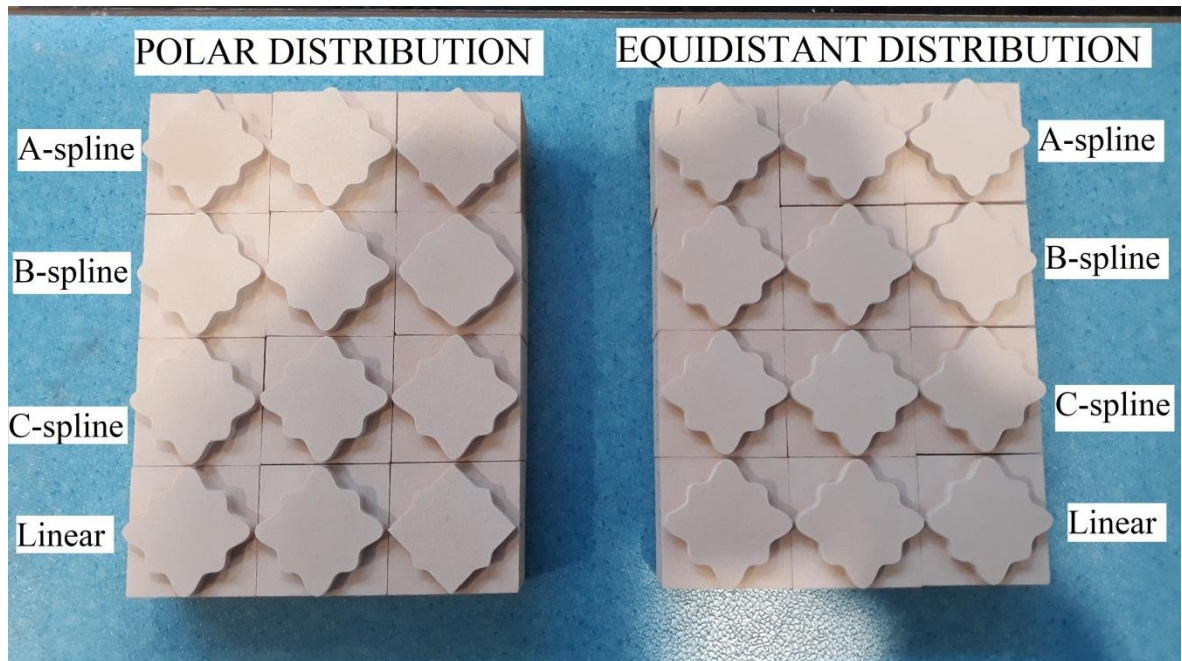


Figure 3.9.2 All machined contours using 3D contouring.

### 3.10 Methodology of evaluating the surface positions

To compare the milled contours, surface positions were compared. Only 3D solutions were compared as using 2D or 3D contouring would result in identical contour with the exception of the first local maxima where the shape of the contour was influenced for 2D contouring as was described previously in 3.8.

The accuracy of machined contours were Eleven control points were chosen in the second quadrant at local maxima and minima and for certain values of steptheta chosen to obtain different points from the knots in the programs. Local maxima in the second quadrant are situated at  $\theta = 90^\circ$  (control point 1),  $\theta = 135^\circ$  (control point 6),  $\theta = 180^\circ$  (control point 11) while the local minima are located at angles  $\theta = 115^\circ$  (control point 4) and  $\theta = 155^\circ$  (control point 8). Control points coordinates are stated in Table 3.10.1 and represented on Figure 3.10.1 and Figure 3.10.2.

Table 3.10.1 Control points.

control point	steptheta = angle (°)	x (mm)	y (mm)
1	90,0	0,000	22,000
2	100,5	-3,589	19,366
3	107,5	-5,158	16,358
4	115,0	-6,666	14,296
5	127,5	-10,512	13,700
6	135,0	-12,728	12,728
7	147,5	-13,804	8,794
8	155,0	-14,296	6,666
9	167,5	-18,489	4,099
10	172,0	-20,373	2,863
11	180,0	-22,000	0,000

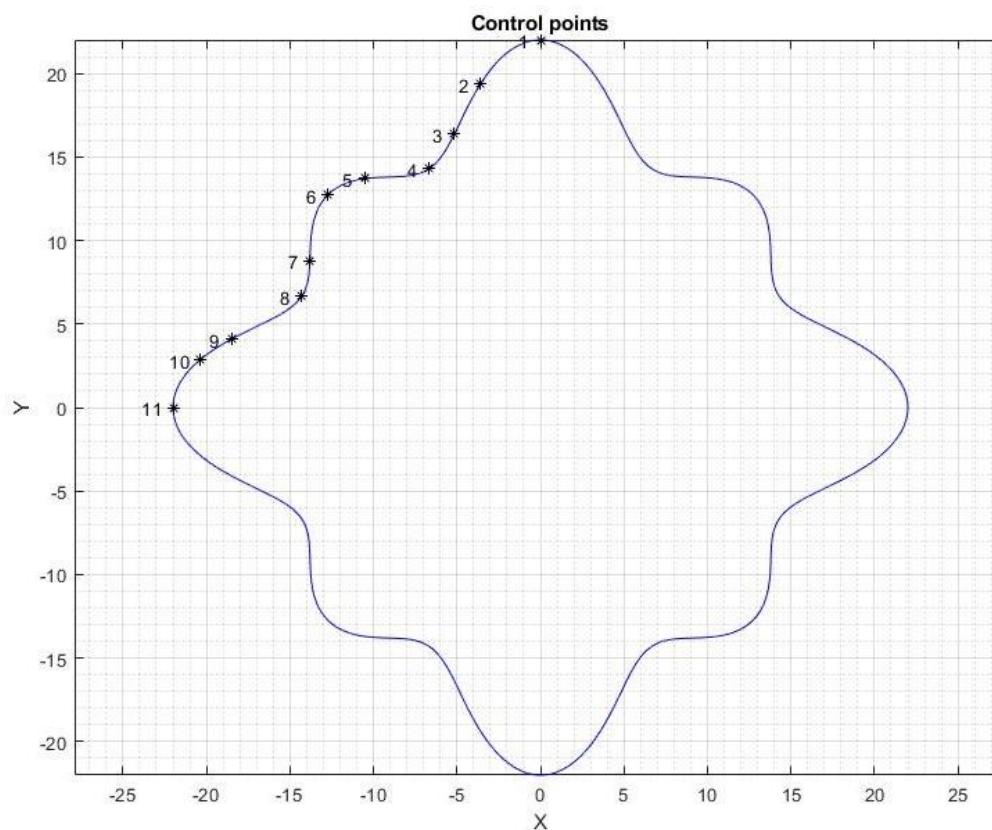


Figure 3.10.1 Control points on the theoretical contour.

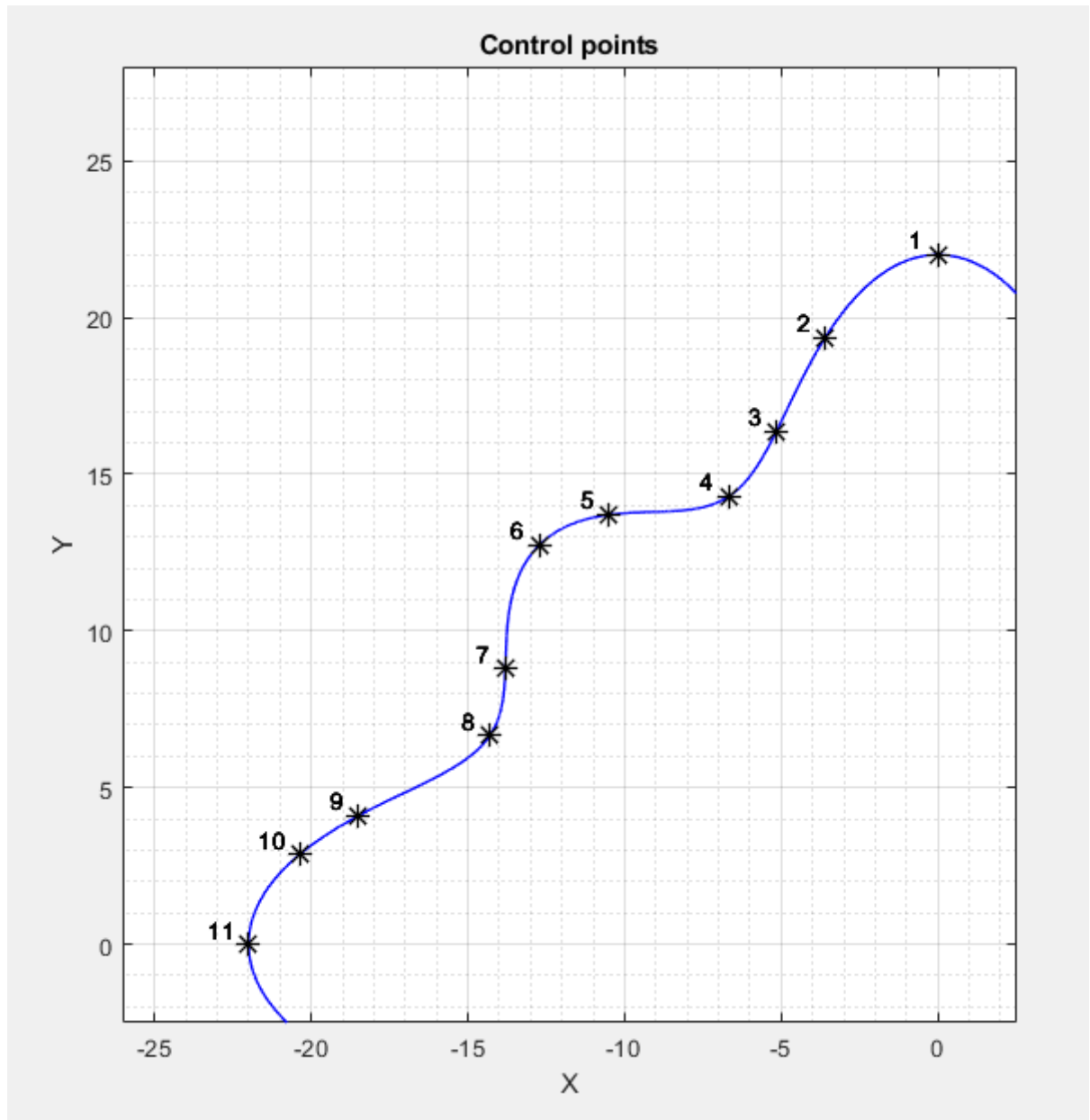


Figure 3.10.2 Control points in the second quadrant on the theoretical curve.

Measurements of the milled surface positions were originally meant to be performed by the same probe as for setting the workpiece zero. It's a built in probe meaning the measurements can be realised right after finishing. Nevertheless the smallest diameter of the probe available at the time in the workplace was 6 mm which is more than the minimal curvature of the contour rendering accurate measurements impossible. It would also be difficult to guarantee that the probe would approach the surface perpendicularly to the surface. The machine's table would have to be rotated meaning the coordinate system would rotate with the clamped workpiece as well. Using the built in probe was therefore not applied.

MarVision MM 420 workshop measuring microscope by company Mahr was used instead to evaluate surface positions in order to compare the milled contours to the theoretical one. MarVision MM 420 offers smallest magnifications of 0,7 and largest magnification 4. The studied part imagery was directly transmitted to the computer as well as values of X and Y coordinated in Cartesian coordinate system that is referred to positions of the table axis X and Y.

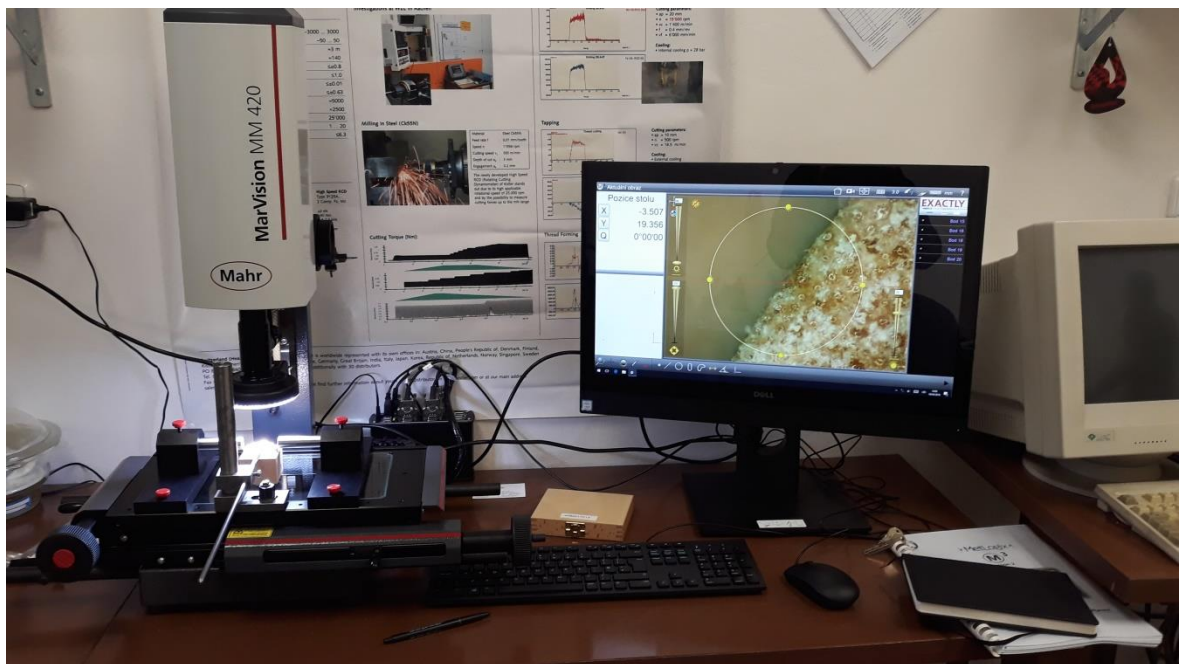


Figure 3.10.3 MarVision MM 420 workshop measuring microscope

Firstly the location of the workpiece was clamped and the workpiece origin was determined in order to be able to take surface positions in the Cartesian coordinate system in which the contour was machined. In the workpiece origin, values of X and Y axis of the table of MarVision were set to 0 value therefore both coordinate systems were identical.

Real surface positions in the second quadrant were then compared to the control points generated previously. Y coordinate of the every control was point fixated and its real X coordinate was verified. How much the real value of its X coordinate was different from the theoretical value was then compared between methods. The only exception was the first control point, where X value was fixated and Y coordinate was verified because it wasn't possible to fixate the Y coordinate as it was different from the theoretical value and there wasn't material of witch to take position of.

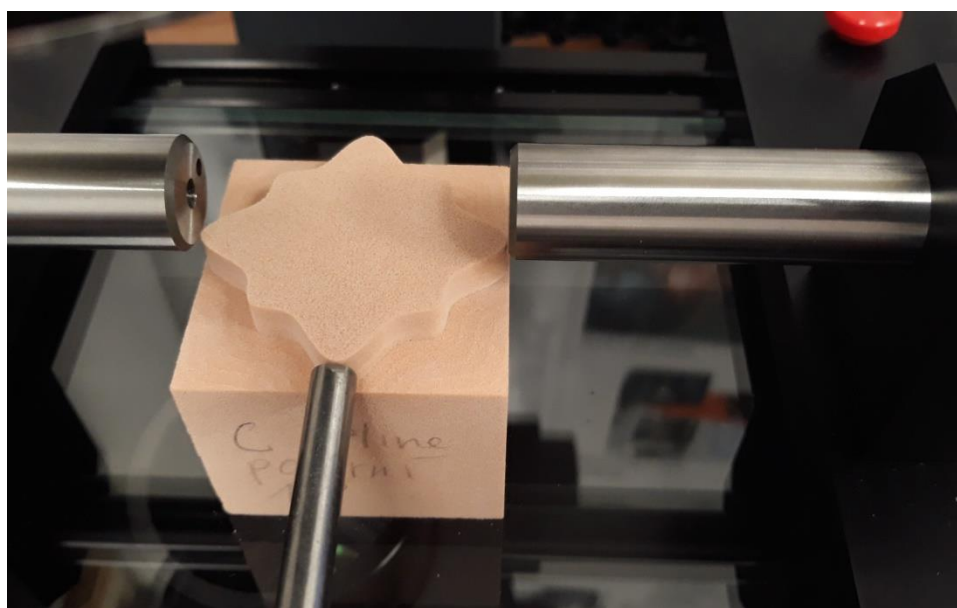
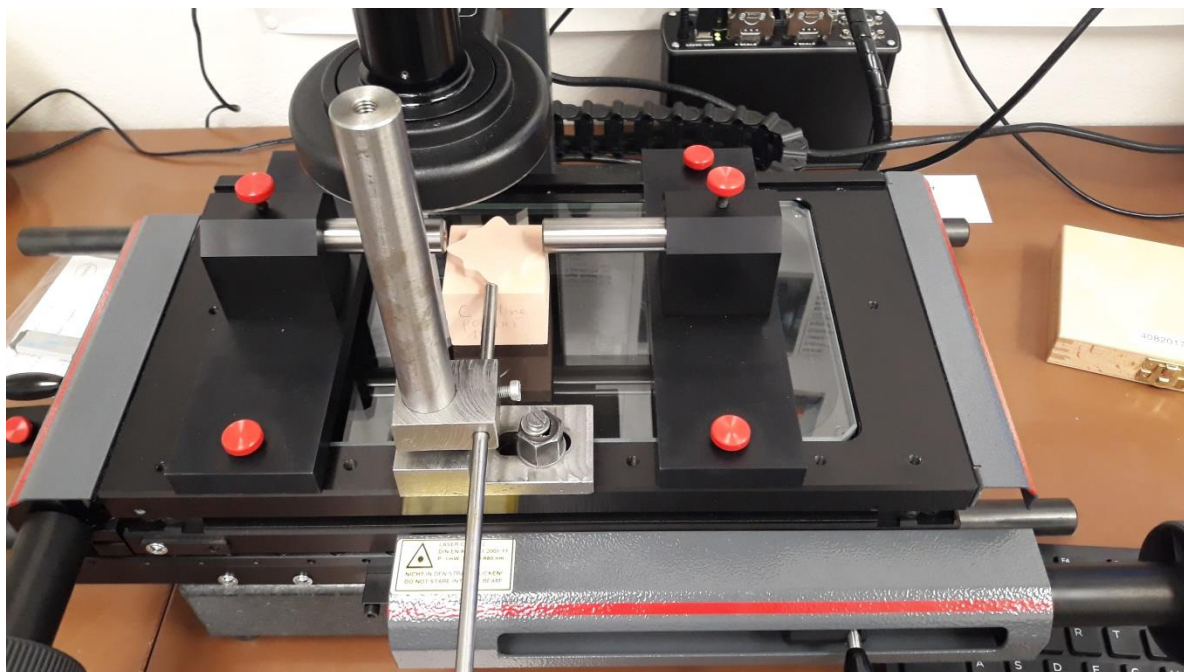

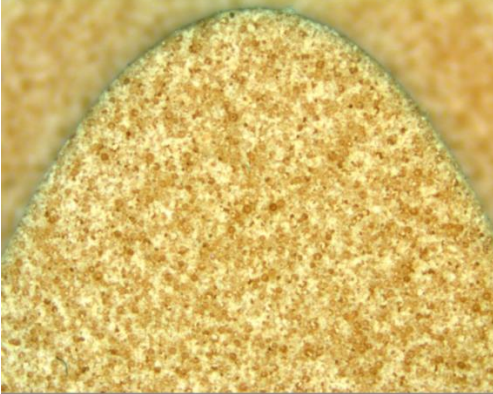
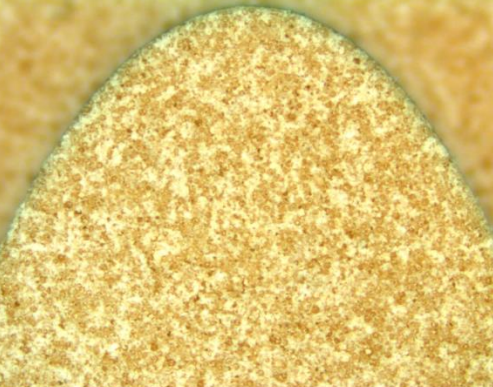



Figure 3.10.4 Clamping of the workpiece in MarVision.

### 3.11 Observations

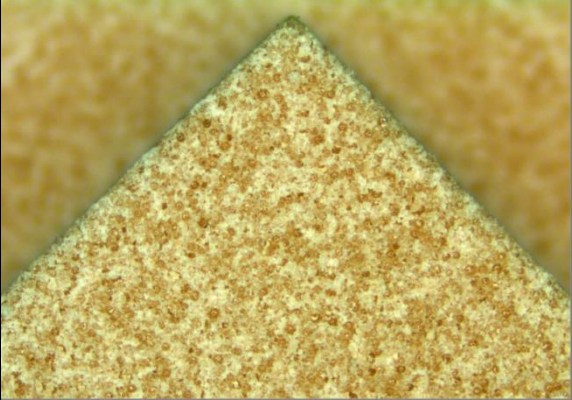

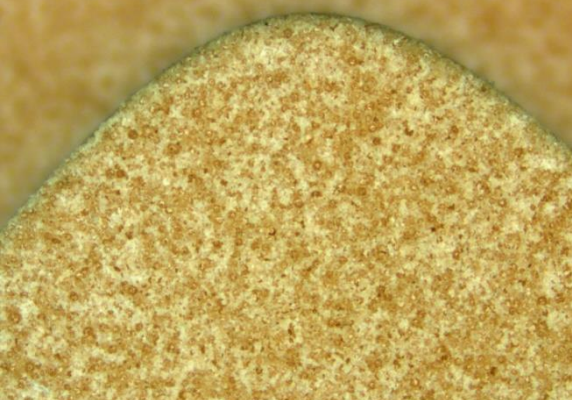

For visualisation, local maxima of workpieces machined using equidistant distribution with 357 knot per contour viewed under microscope are displayed side by side for all used interpolation method. In this case all methods were provided sufficient number of knots and there wasn't note any remarkable difference of surface positions as visible in on figures in Table 3.11.1.

Table 3.11.1 Comparison of performance of spline interpolation methods in vicinity of the first local maxima for equidistant distribution with total of 357 knots.

Linear, equidistant distribution, 357 knots	A-spline, equidistant distribution, 357 knots
	
B-spline, equidistant distribution, 357 knots	C-spline, equidistant distribution, 357 knots
	

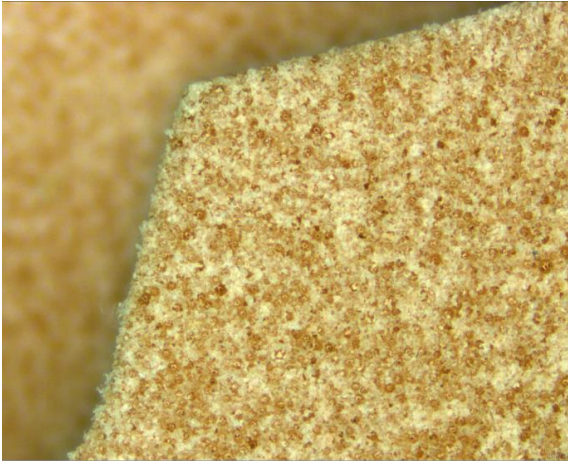
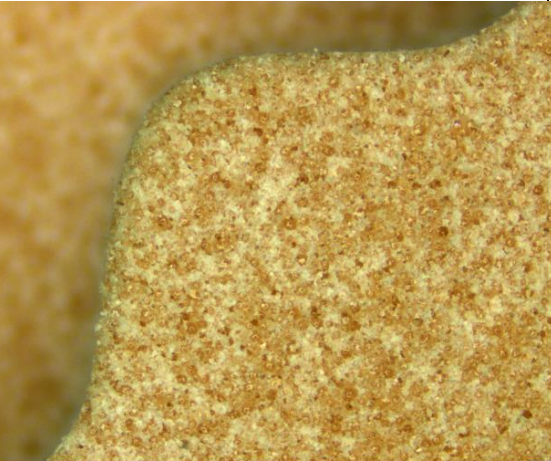


On the other hand the version with least points out of all applied varieties was polar distribution with  $\text{step}_{\theta} = 15^{\circ}$  with only 24 knots per contour. It shows that the only method which usage resulted in machining a contour closest to the theoretical contour was C-spline interpolation. First local maxima obtained by this distribution of knots are displayed in Table 3.11.2.

Table 3.11.2 Comparison of performance of spline interpolation methods in vicinity of the first local maxima.

Linear, polar distribution, 24 knots	A-spline, polar distribution, 24 knots
	
B-spline, polar distribution, 24 knots	C-spline, polar distribution, 24 knots
	

Second local maxima for this distribution milled using different spline methods is displayed in Table 3.11.3.

Table 3.11.3 Comparison of performance of spline interpolation methods in vicinity of the second local maxima.

Linear, polar distribution, 24 knots	A-spline, polar distribution, 24 knots
 A close-up photograph of a textured surface, possibly a material sample, showing a sharp corner. The image is rendered using linear interpolation with 24 knots, resulting in a visible staircase effect or aliasing artifacts along the curved boundary.	 A close-up photograph of the same textured surface and corner, rendered using A-spline interpolation with 24 knots. The curved boundary is significantly smoother and more continuous compared to the linear method.
B-spline, polar distribution, 24 knots	C-spline, polar distribution, 24 knots
 A close-up photograph of the textured surface and corner, rendered using B-spline interpolation with 24 knots. The boundary is smooth and well-defined.	 A close-up photograph of the textured surface and corner, rendered using C-spline interpolation with 24 knots. The boundary is smooth and well-defined, similar to the B-spline result.



The differences between spline interpolation methods for polar distribution with 24 knots ( $\text{step}_{\theta} = 15^{\circ}$ ) were very visible even by naked eye as illustrated on Figure 3.11.1 and demonstrate well the limits of each spline interpolation method.

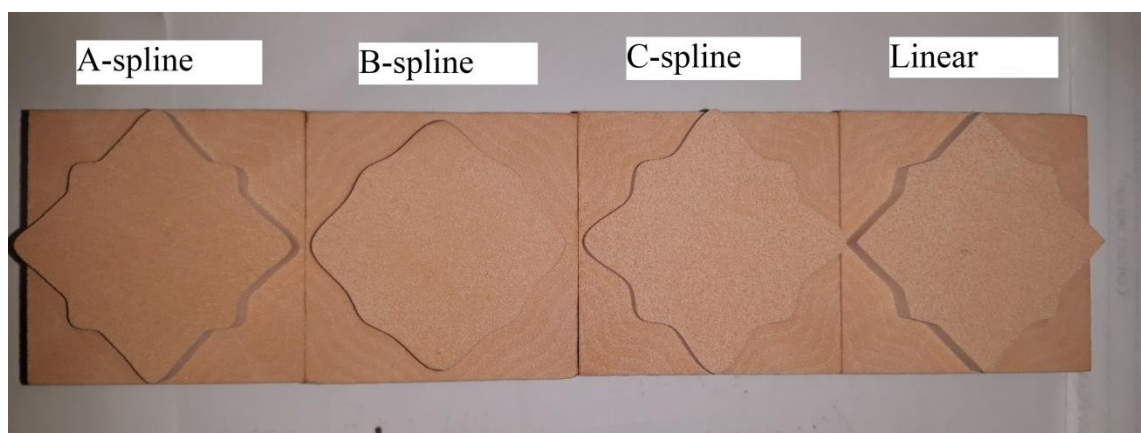
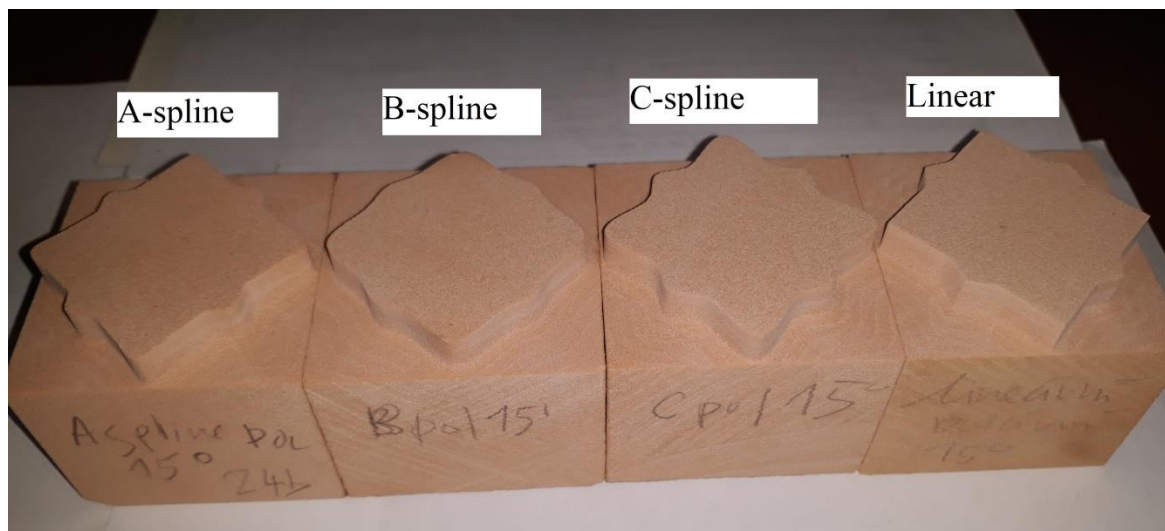


Figure 3.11.1 Milled contours realised by spline interpolations for polar distribution of knots, 24 knots per contour.

When evaluating accuracy of linear interpolation method control points that are at the same time local maxima or minima are identical with some of the knots. As linear interpolation passes directly through the knots, it might augment its statistics about its accuracy. B-spline does not pass through the knots therefore its inaccuracy in those control points was seemingly bigger than that of a linear interpolation method.

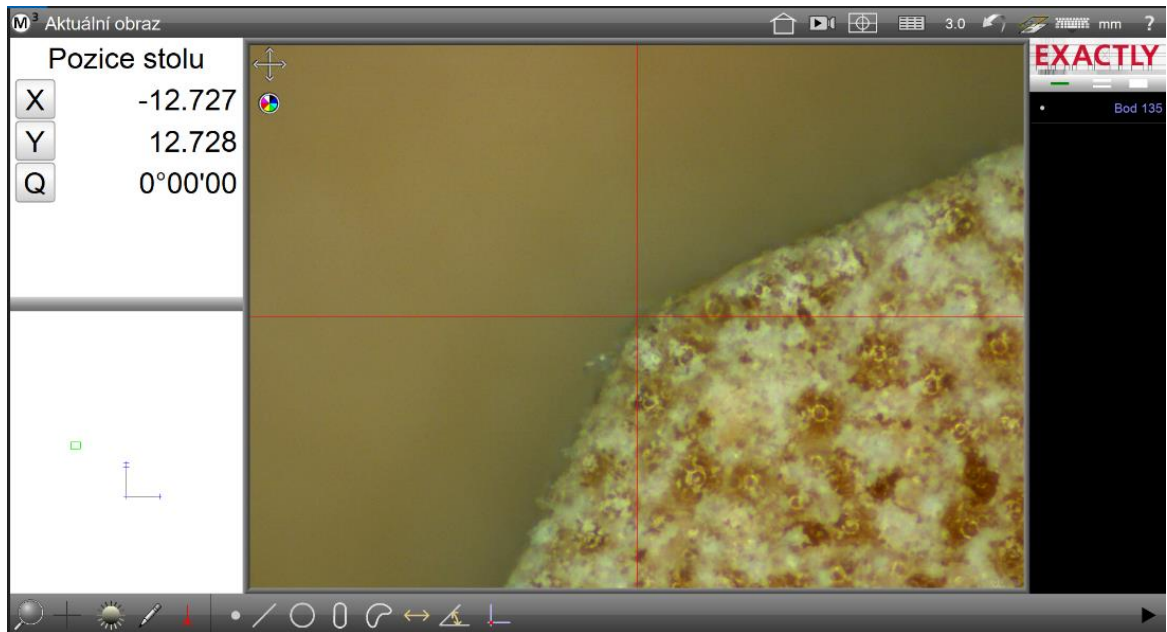


Figure 3.11.2 Linear interpolation passing directly through a knot.

Linear interpolation resulted in formation of facets as viewed under microscope on Figure 3.11.3 and Figure 3.11.4. The facets are also visible with naked eye as is illustrated on Figure 3.11.5.

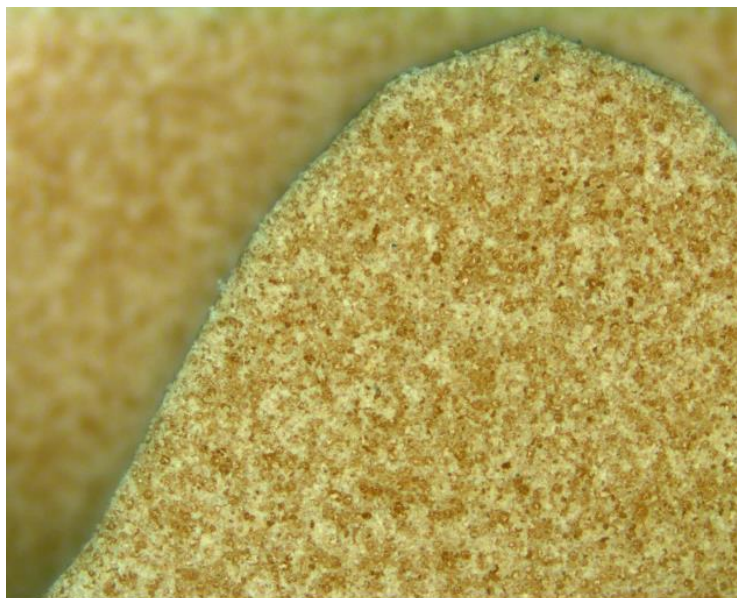


Figure 3.11.3 Linear, polar distribution, 24 knots, facets in second quadrant, first local maximum.

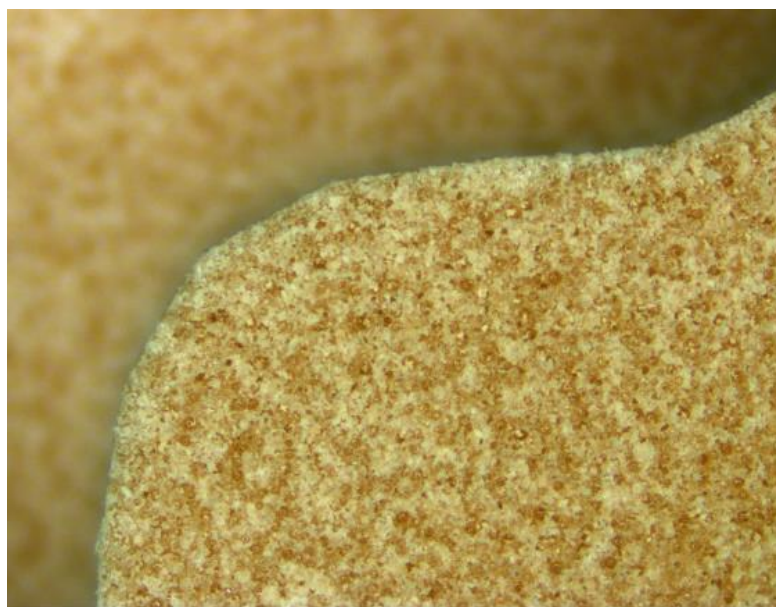


Figure 3.11.4 Linear, polar distribution, 24 knots, facets in second quadrant, second local maximum.

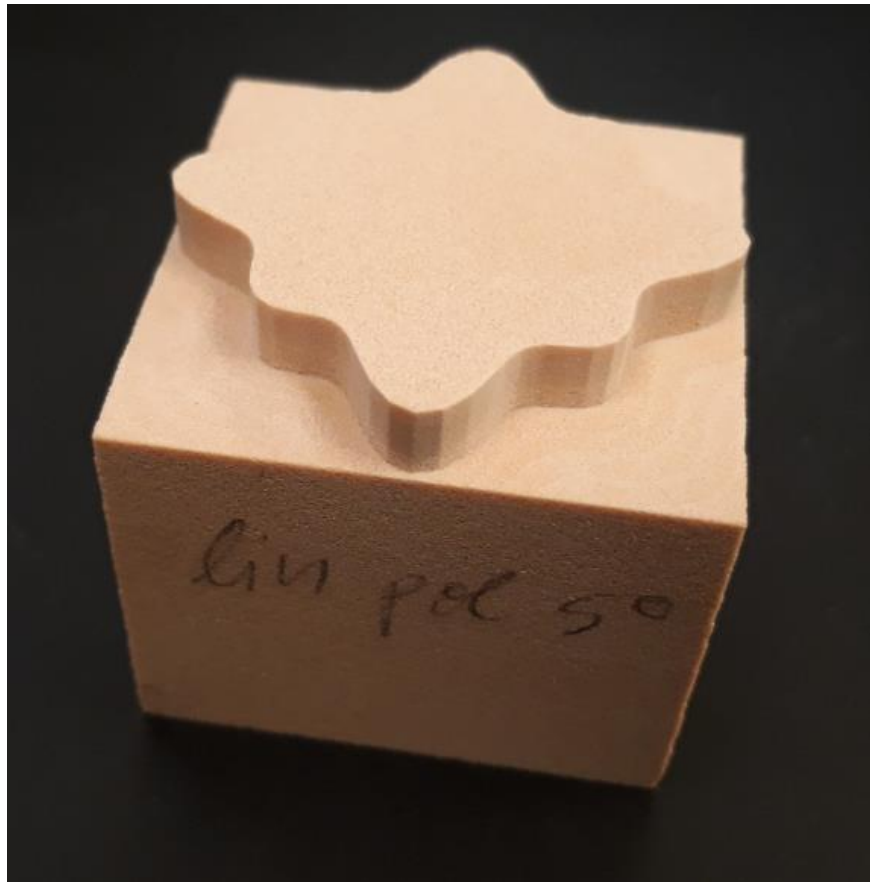


Figure 3.11.5 Visible facets on contour created using linear interpolation, polar distribution of 72 knots.

On the other hand, all employed spline interpolation resulted in smooth curves. Even though the contour was not precise, it still was a smooth curve with better surface roughness.

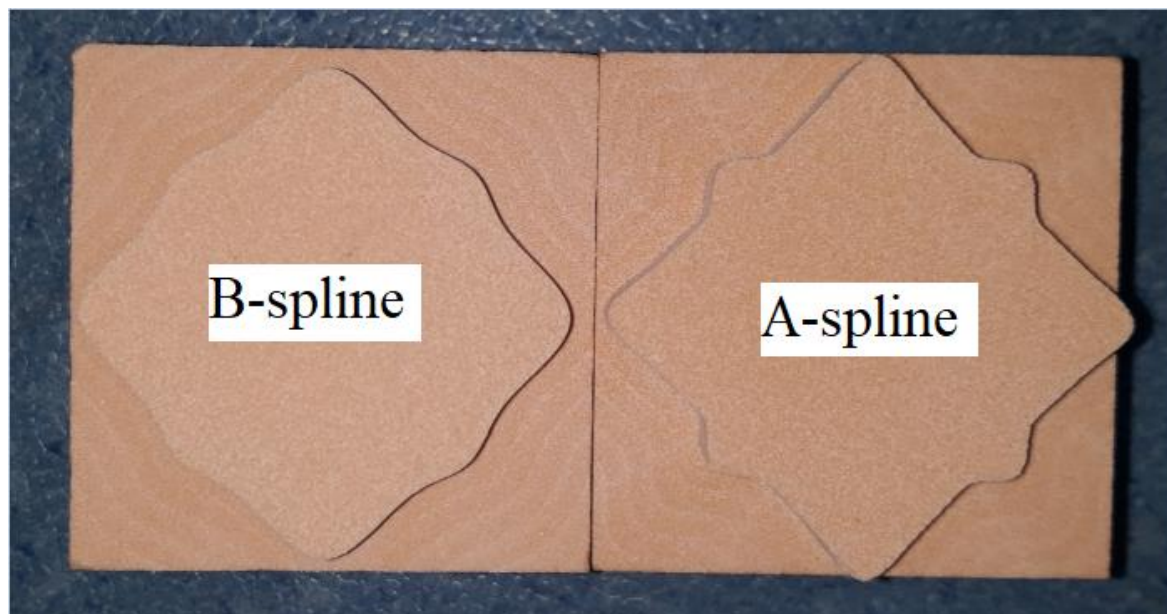


Figure 3.11.6 B-spline and A-spline interpolation for polar distribution with 24 knot per contour.

Following charts in 3.12.3 and 3.12.4 represent difference between theoretical coordinate and measured coordinate for every control point from 1 to 11. For theoretical curve the difference is equal to zero and represents perfect correspondence. Error is marked on radial axis in millimetres.

### 3.12 Interpretation of measurements of surface position

Difference of the machined surface coordinate and control point coordinate generated in 3.10 and was measured for all 24 workpieces. Error in individual points and overall accuracy in form of absolute mean error were calculated.

#### 3.12.1 Error in individual control points

Error on individual points was calculated by subtracting the measured control point's coordinate from its theoretical value. The errors were then inserted in polar charts that displays if the control point coordinate is located further from the origin or closer to the origin of the Cartesian coordinate system of the workpiece.

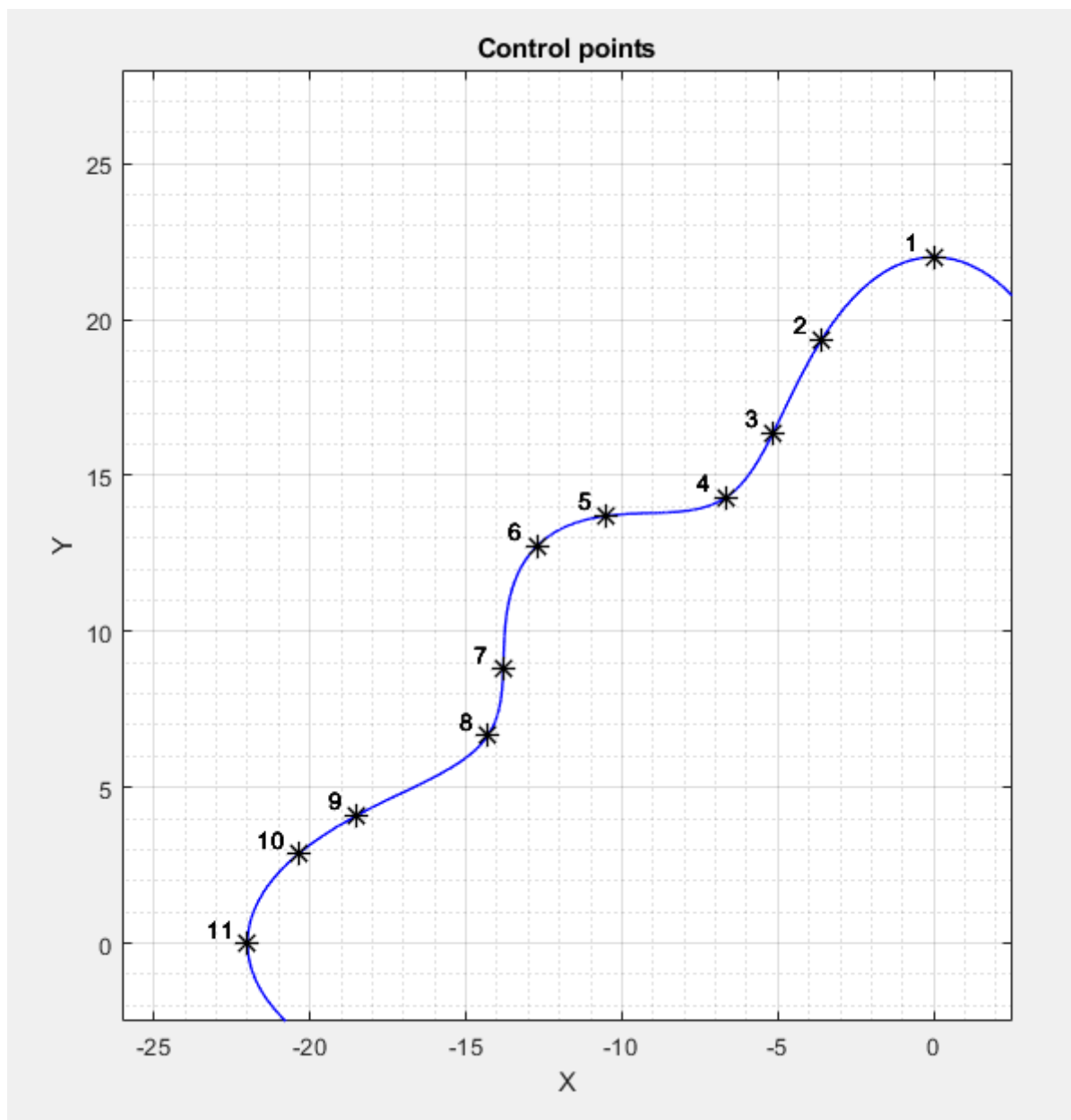


Figure 3.12.1.1 Control points in second quadrant.

The numbers of control points are on labelled on perimeter and the error is displayed in millimetres on the radial axis. Zero error would be represented as on Figure 3.12.1.2.

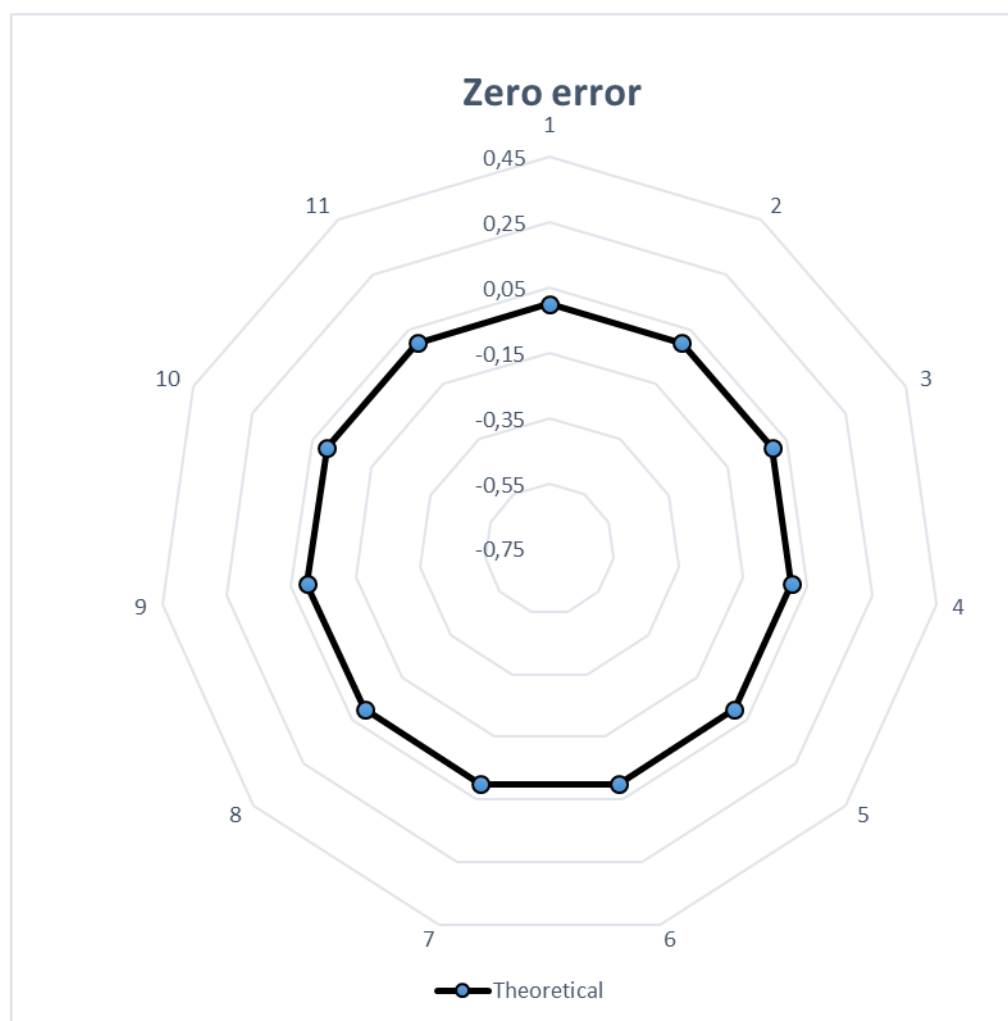


Figure 3.12.1.2 Representation of zero error of control points coordinates.

### 3.12.2 Absolute mean error of methods

To statistically evaluate the overall accuracy of spline interpolation methods, mean absolute error was calculated for all employed methods. Mean absolute error (MAE) is the average of all absolute errors and is defined as (37) [30].

$$MAE = \frac{1}{N} \sum_{i=1}^N |x_i - x| \quad (37)$$

Where the abbreviations in (37) stand for [30]:

N ... number of measured values

$|x_i - x|$  ... absolute error

x ... real value of x

$x_i$  ... measured value of x

### 3.12.3 Accuracy of spline interpolations for polar distribution

#### 3.12.3.1 Error of spline interpolations for polar distribution in individual control points

Measured errors of control point coordinates obtained using different spline interpolation methods with polar distribution of knots are visualised on Figure 3.12.3.1.1, Figure 3.12.3.1.2 and Figure 3.12.3.1.3.

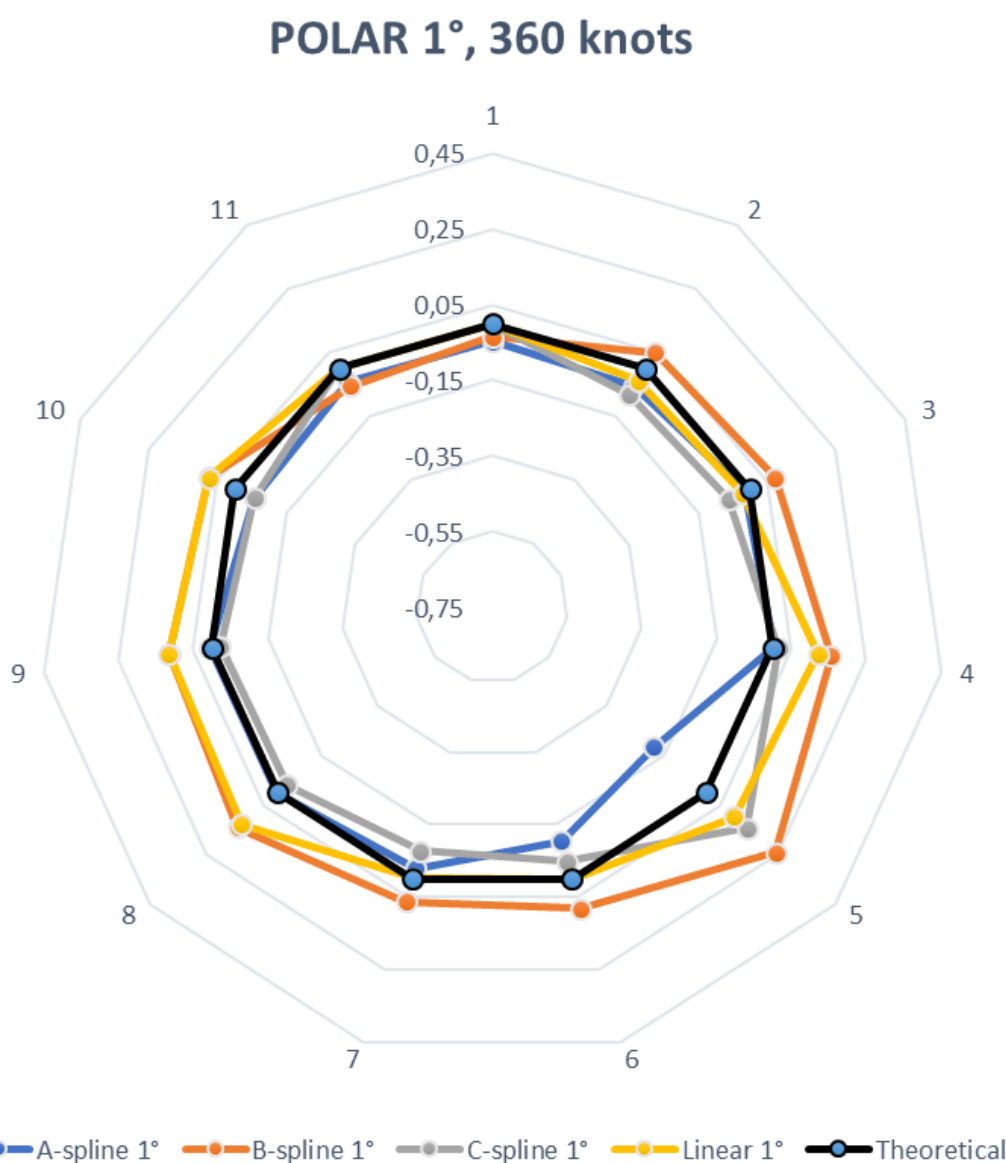


Figure 3.12.3.1.1 Error coordinates of control points, polar distribution of 360 knots.



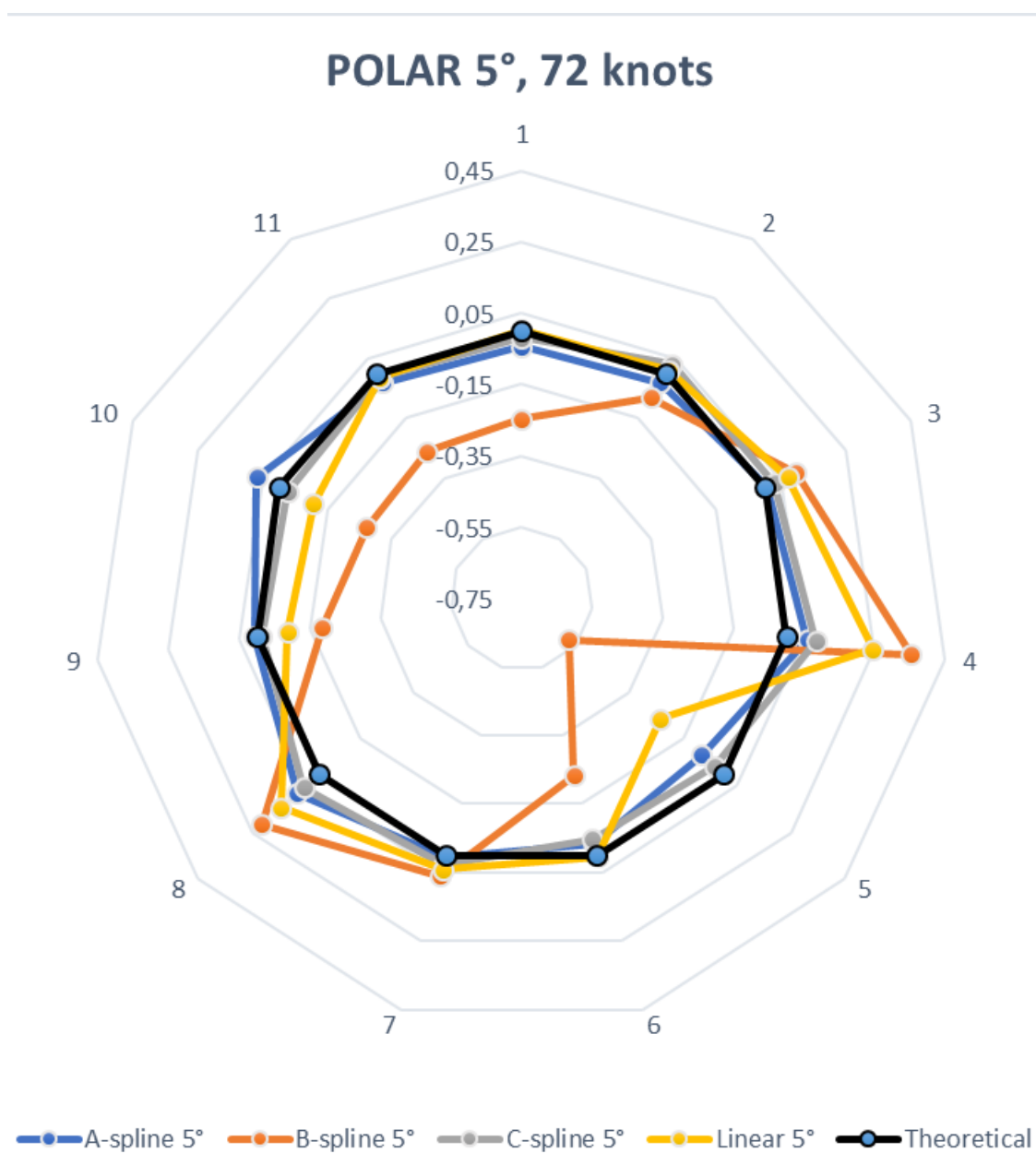


Figure 3.12.3.1.2 Error of coordinates of control points, polar distribution of 72 knots.

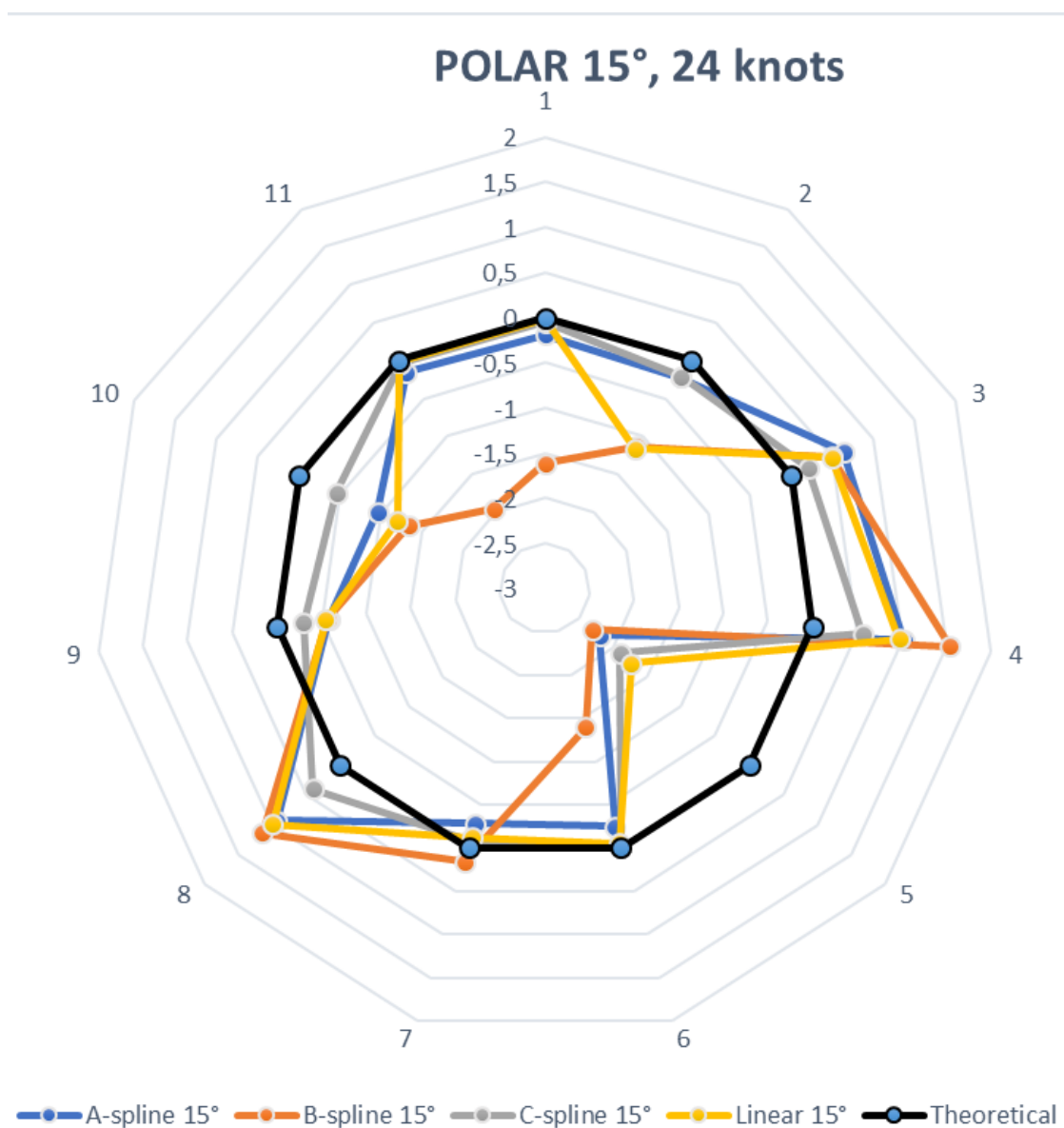


Figure 3.12.3.1.3 Error of coordinate of control points, polar distribution of 24 knots.

### 3.12.3.2 Mean absolute error of spline interpolations for polar distribution

The mean of absolute values of errors of coordinates in for chosen interpolation method for all used equidistant distributions of knots are compared on Figure 3.12.3.2..

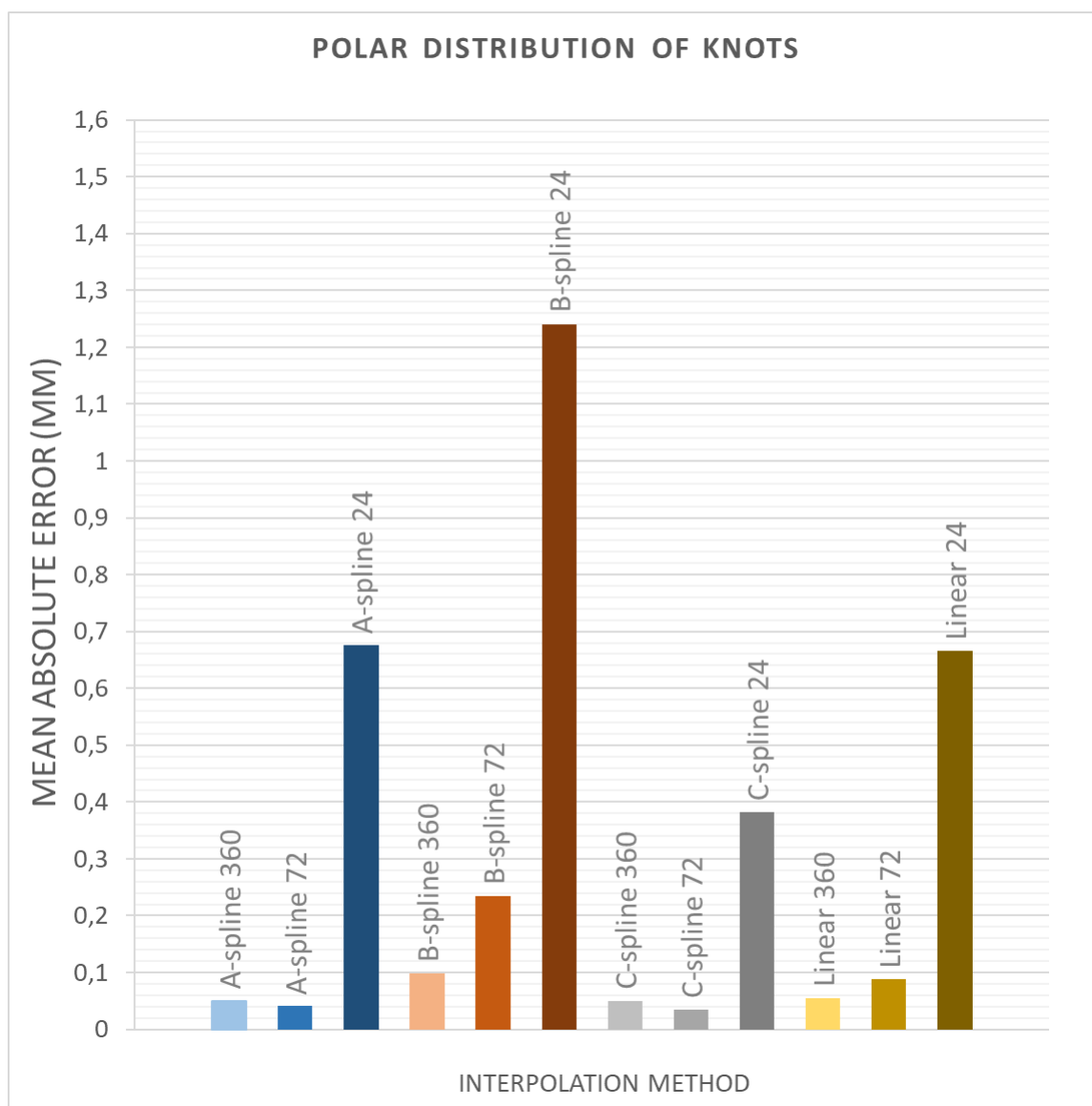


Figure 3.12.3.2. Comparison of mean absolute error of spline interpolation method for polar distribution of knots.

The mean of absolute values of errors of coordinates in for employed interpolation methods for polar distribution of 360 knots are compared on Figure 3.12.3.4.

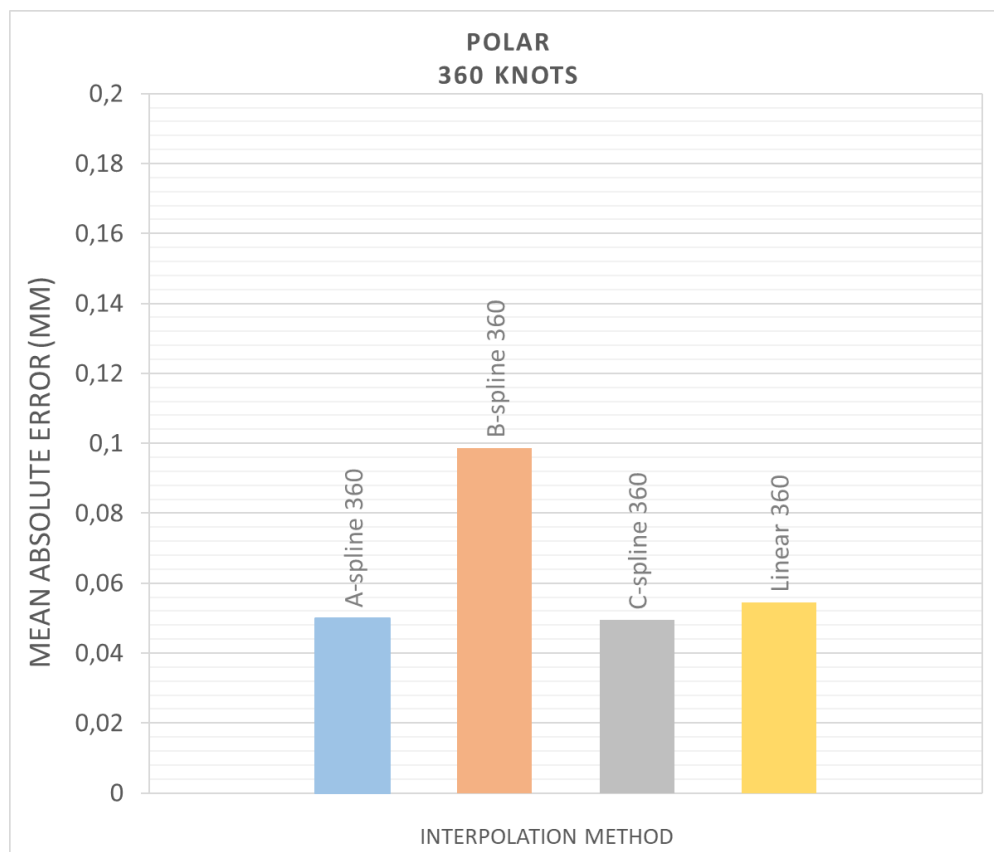


Figure 3.12.3.4 Mean absolute error of spline interpolations for polar distribution of 360 knots.

The mean of absolute values of errors of coordinates in for employed interpolation methods for polar distribution of 72 knots are compared on Figure 3.12.3.5.

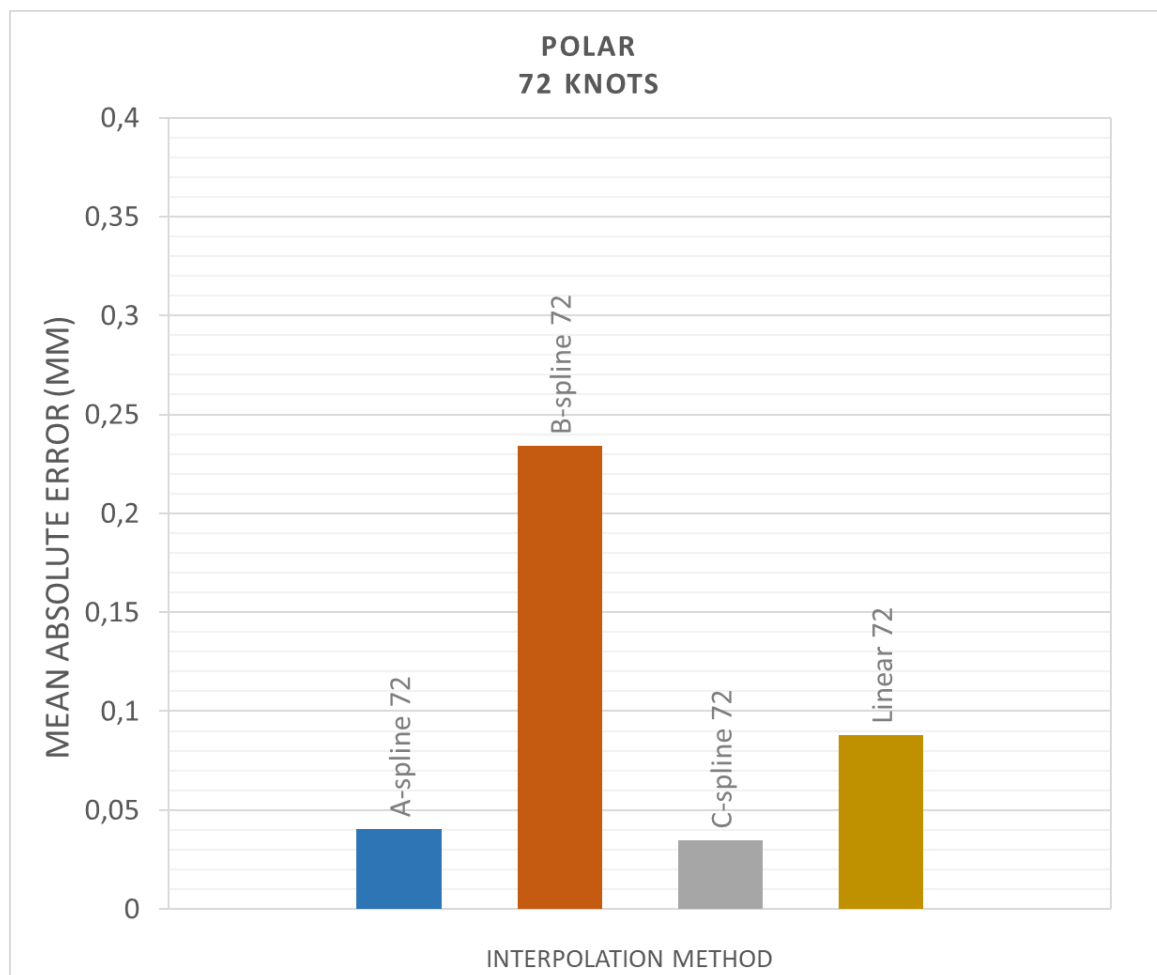


Figure 3.12.3.5 Mean absolute error of spline interpolations for polar distribution of 72 knots.

The mean of absolute values of errors of coordinates in for employed interpolation methods for polar distribution of 24 knots are compared on Figure 3.12.3.6.

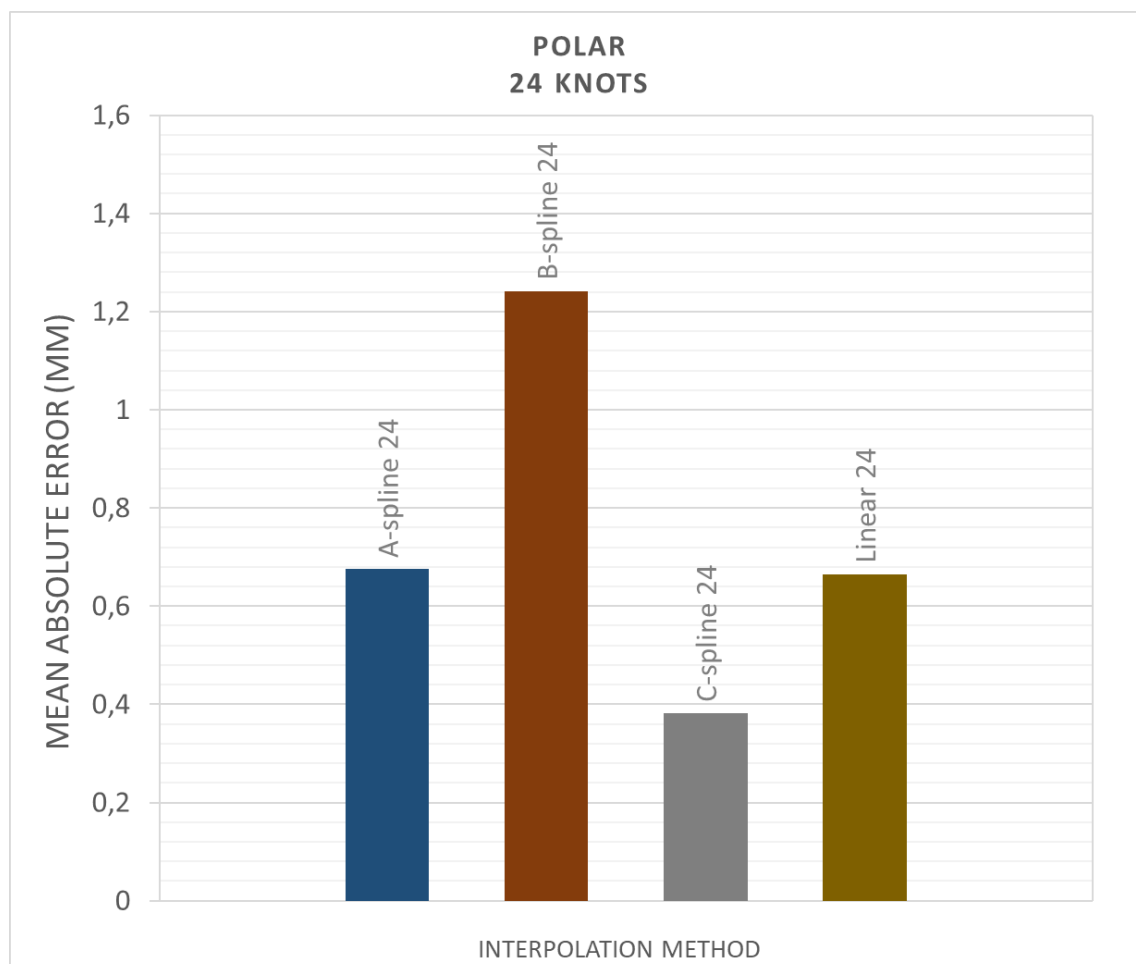


Figure 3.12.3.6 Mean absolute error of spline interpolations for polar distribution of 24 knots.

### 3.12.4 Accuracy of spline interpolations for equidistant distribution

#### 3.12.4.1 Error of spline interpolations for equidistant distribution in individual control points

Measured error of control points coordinates for equidistant distribution of knots are visualised on Figure 3.12.4.1, Figure 3.12.4.2 and Figure 3.12.4.3.

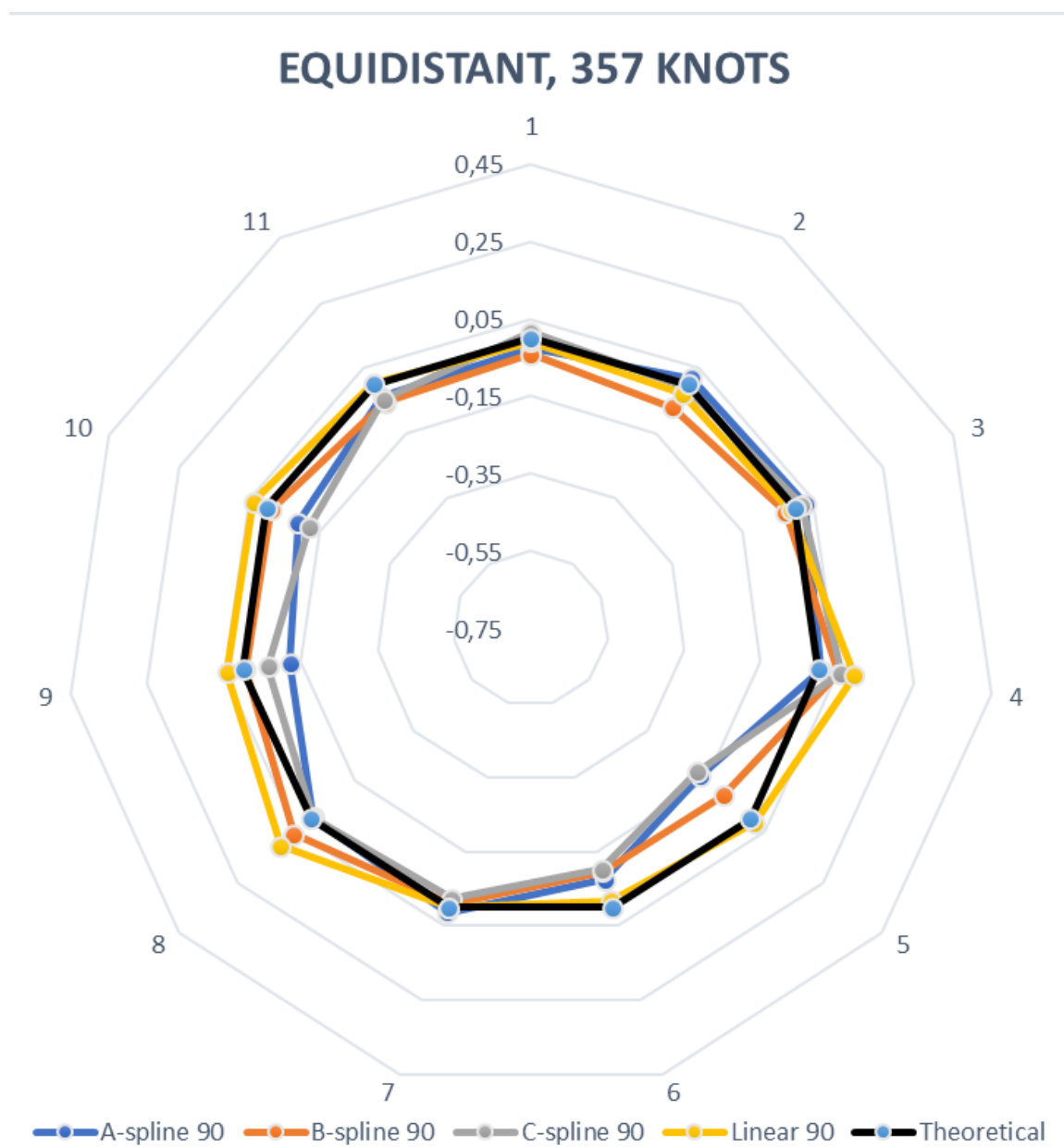


Figure 3.12.4.1 Difference of coordinates of control points, equidistant distribution of 357 knots.

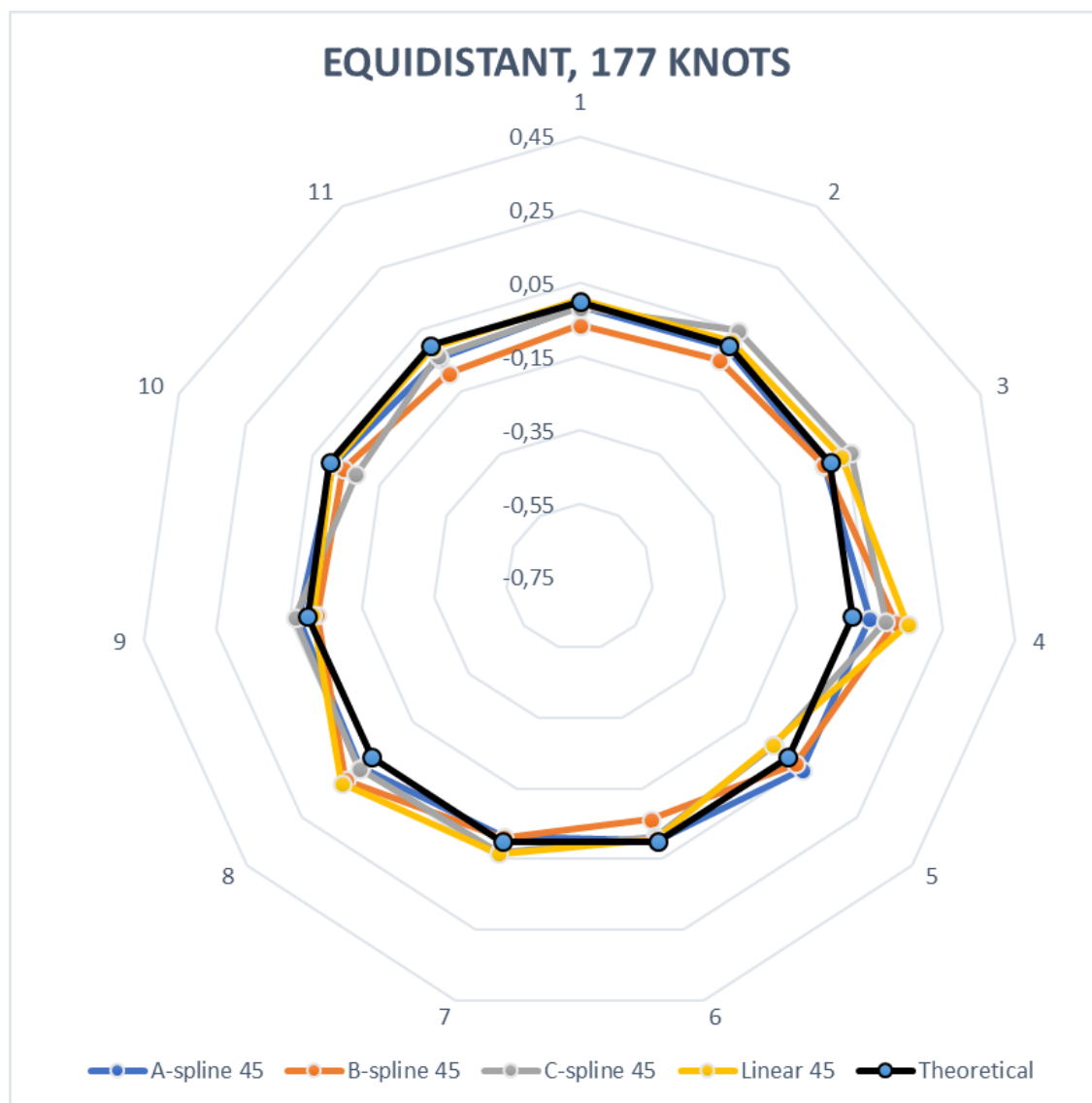


Figure 3.12.4.2 Difference of coordinates of control points, equidistant distribution of 177 knots.



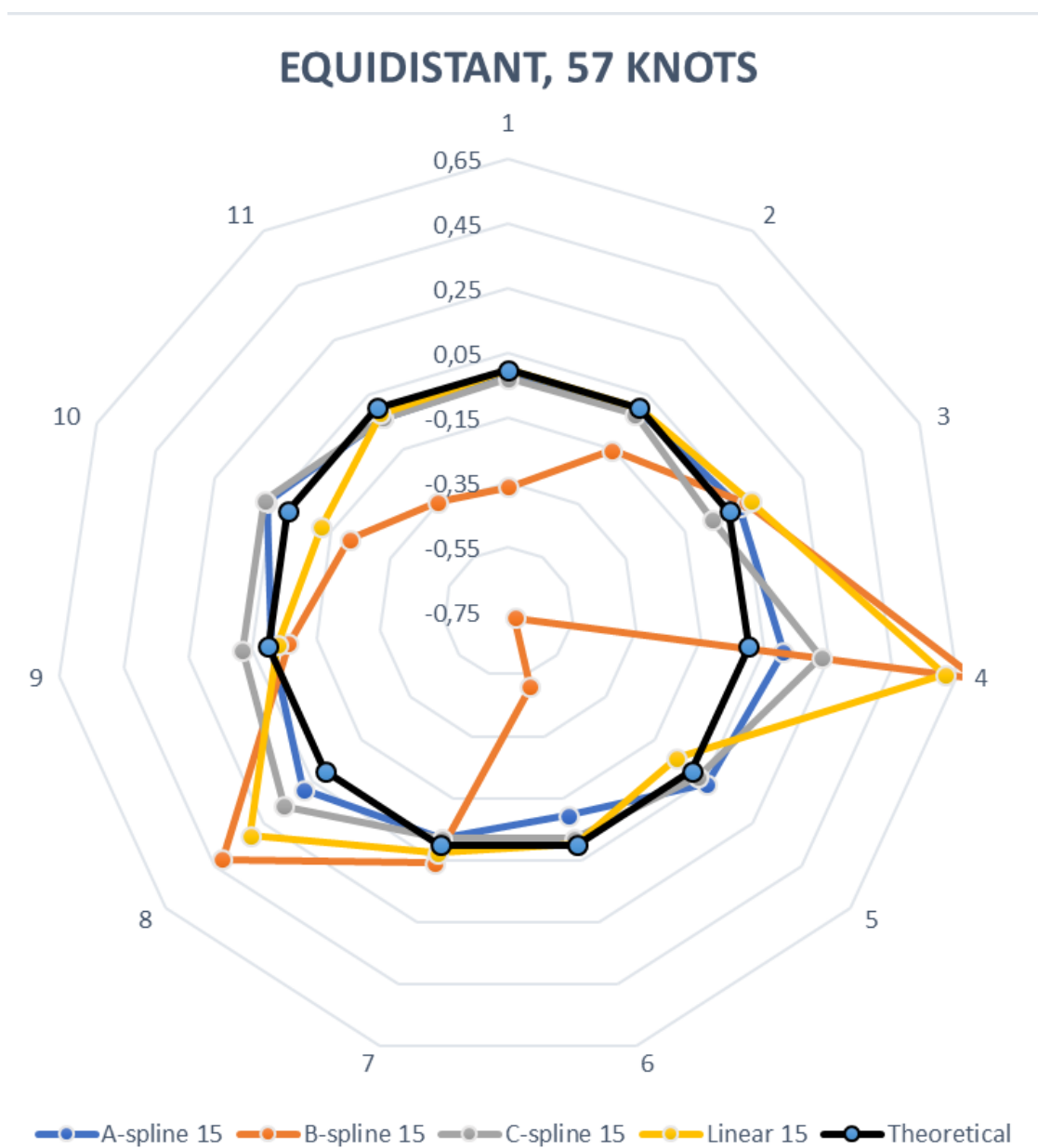


Figure 3.12.4.3 Difference of coordinates of control points, equidistant distribution of 357 knots.

### 3.12.4.2 Mean absolute error of spline interpolations for equidistant distribution

The mean of absolute values of errors of coordinates in for chosen interpolation method for all used equidistant distributions of knots are compared on Figure 3.12.4.4.

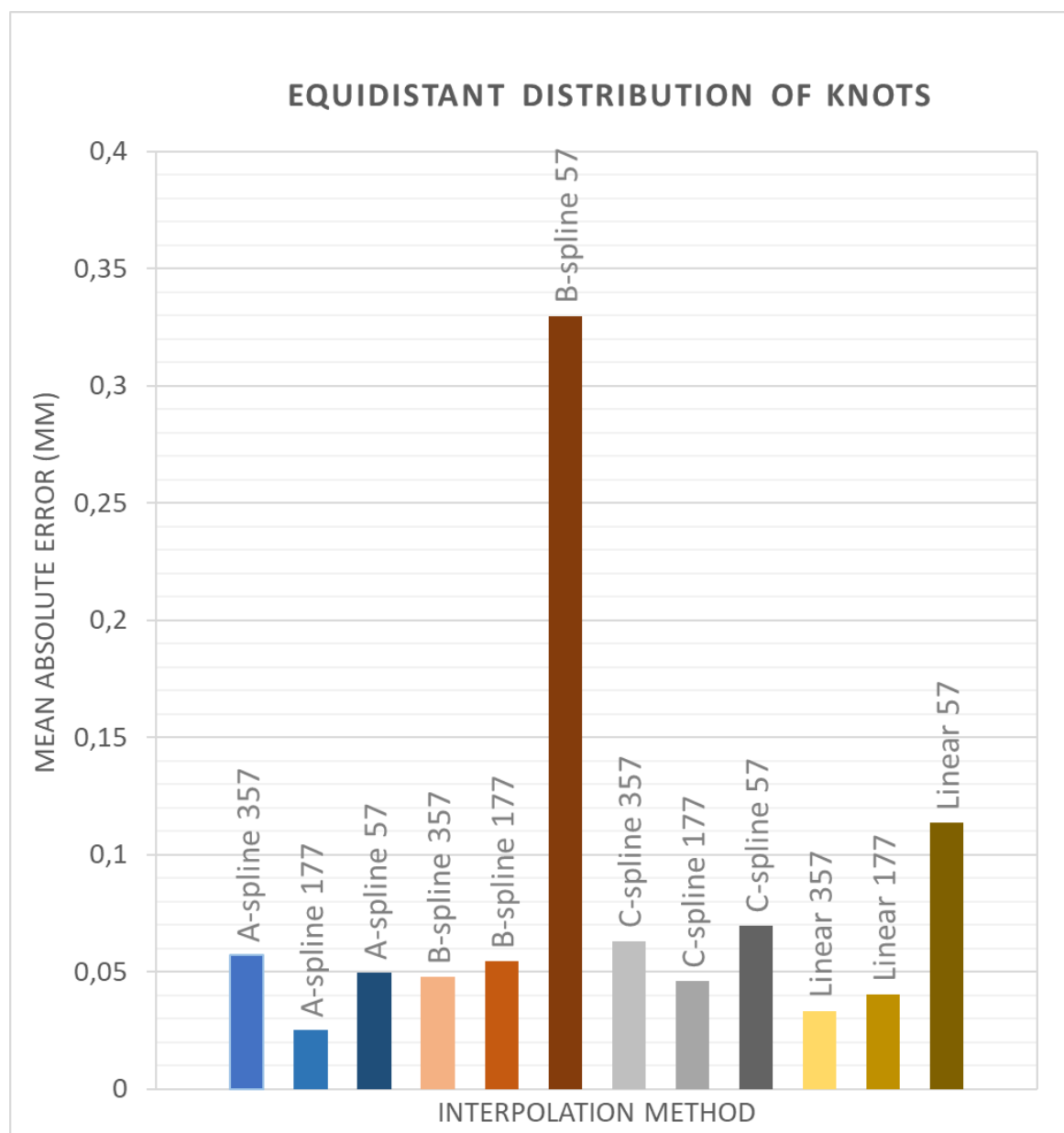


Figure 3.12.4.4 Comparison of mean absolute error of spline interpolation method for equidistant distribution of knots.

The mean of absolute values of errors of coordinates in for employed interpolation methods for equidistant distribution of 357 knots are compared on Figure 3.12.4.5.

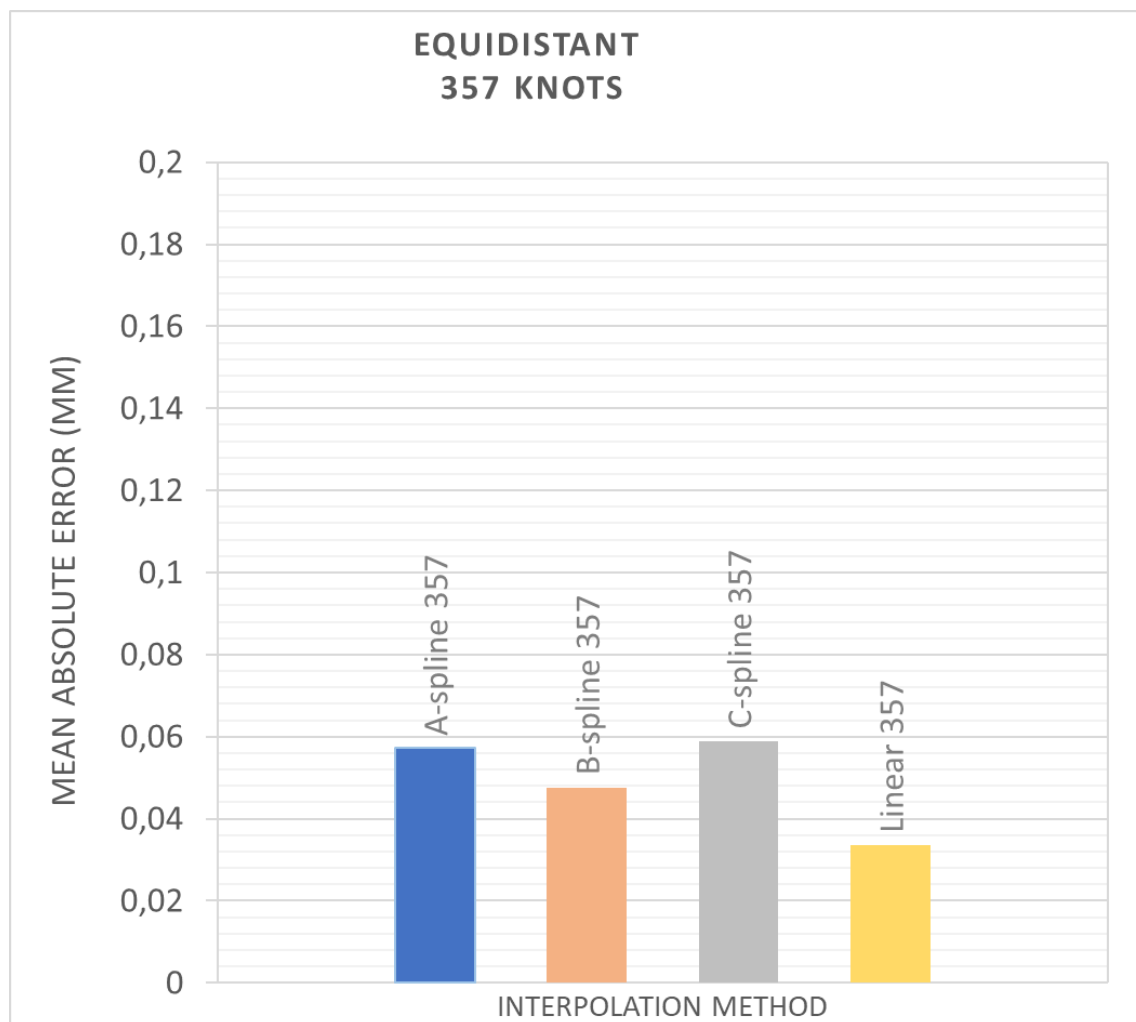


Figure 3.12.4.5 Mean absolute error of spline interpolations for equidistant distribution of 357 knots.

The mean of absolute values of errors of coordinates in for employed interpolation methods for equidistant distribution of 177 knots are compared on Figure 3.12.4.6.

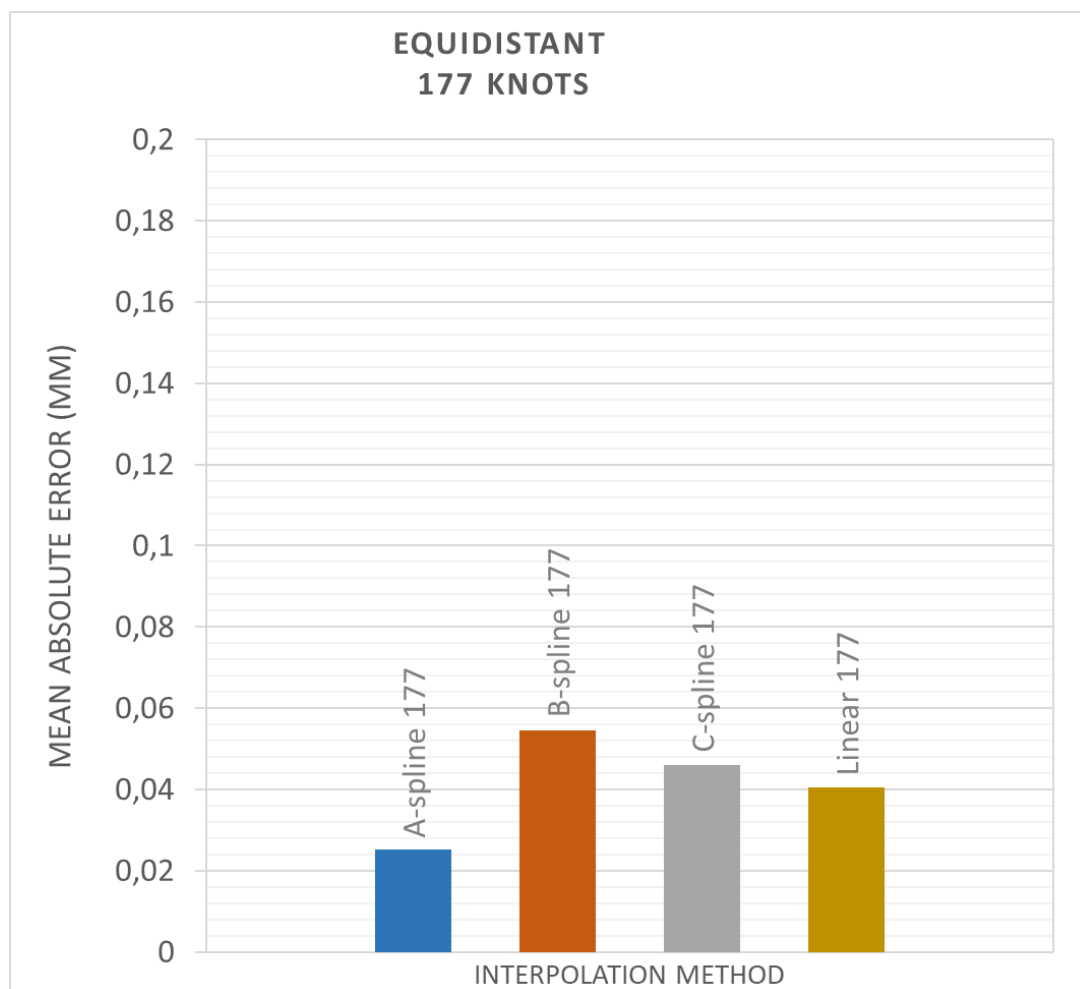


Figure 3.12.4.6 Mean absolute error of spline interpolations for equidistant distribution of 177 knots.

The mean of absolute values of errors of coordinates in for employed interpolation methods for equidistant distribution of 57 knots are compared on Figure 3.12.4.7.

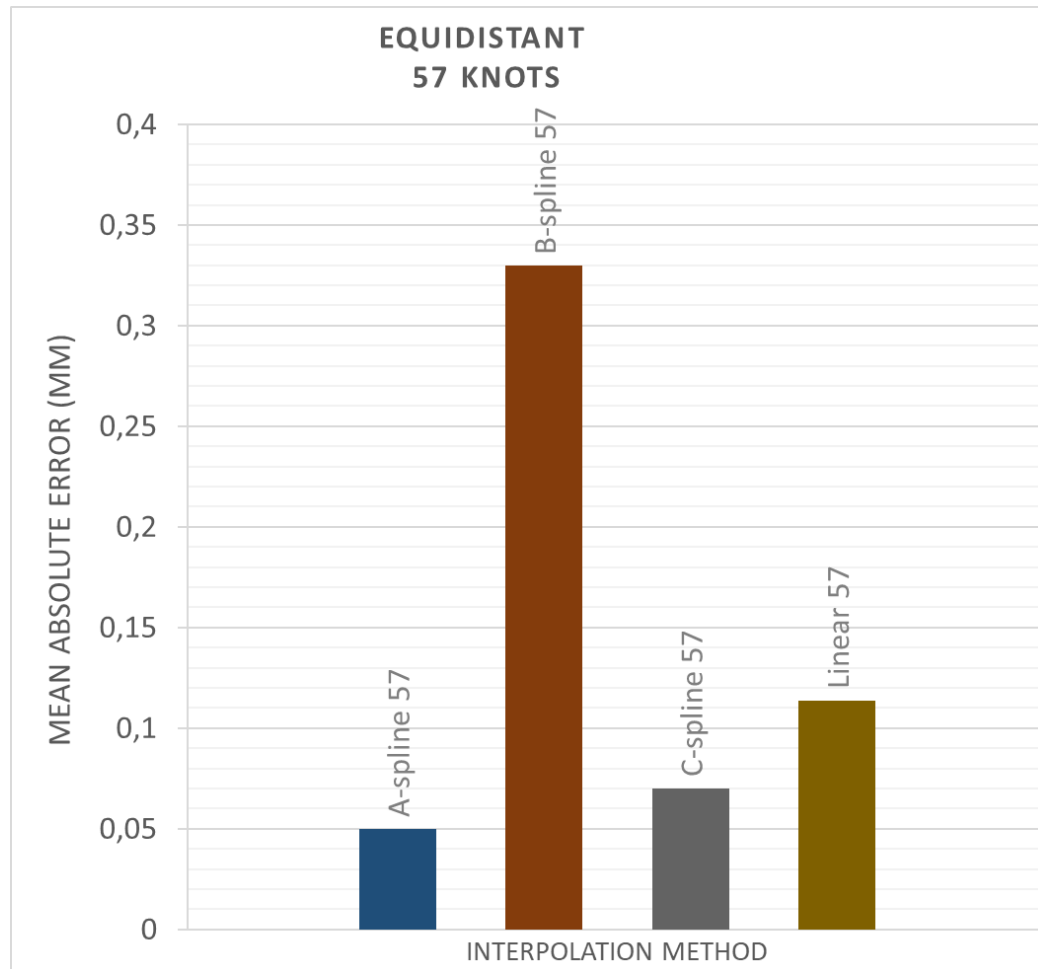


Figure 3.12.4.7 Mean absolute error of spline interpolations for equidistant distribution of 57 knots.

### 3.12.5 Imperfection of used evaluation method

For versions with the highest number of knots, all methods were equally performant and detected error was purely fault of imprecision of used evaluating method.

The employed surface position evaluation method contained some degree of inaccuracy for multiple reasons:

- clamping of the workpiece when using the MarVision was imprecise and only a slight turn of the workpiece would cause inaccuracy, as the coordinates system of the workpiece and of the microscope's table would not be identical,
- only 11 control points were compared,
- human error played a key role,
- some of the control point's location made the evaluation harder and more susceptible to human error.

For example control point number 5 caused increase in error because it was difficult to interpret its real position under the microscope and the slightest difference between coordinate systems would cause grave imprecision as illustrated on Figure 3.12.5.1.

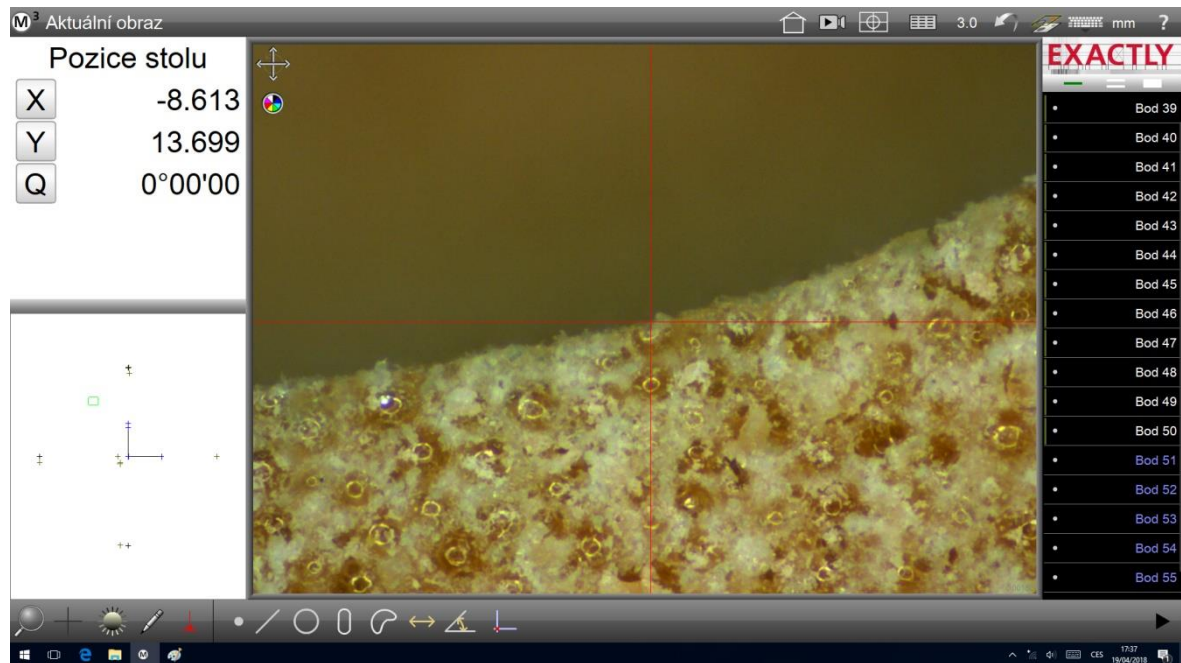


Figure 3.12.5.1 Fifth control point verification for C-spline, polar distribution of knots, stepheta 15°.

### 3.13 Influence of spline interpolation method on machining time

Machining time of execution of finishing was measured for every method.

Even though feed rate was set as a constant, real feedrate was changing its value during the execution. (it was displayed on the control station screen during the execution) Minimal value of real feed rate was noted for every method.

Table 3.13.1 Machining time and minimal value of real feedrate for different spline interpolation methods.

Method	POLAR 360 knots	Linear	A-spline	B-spline	C-spline
	Machining time (s)	27,17	22,34	26,04	28,28
	Minimal Feedrate (mm/min)				
Method	POLAR 72 knots	Linear	A-spline	B-spline	C-spline
	Machining time (s)	37	21,2	19,47	20,65
	Minimal Feedrate (mm/min)	oscilations			
Method	POLAR 24s	Linear	A-spline	B-spline	C-spline
	Machining time (s)	20,28	18,27	15,8	17,93
	Minimal Feed rate (mm/min)	650	oscilations		
Method	EQUIDISTANT 357	Linear	A-spline	B-spline	C-spline
	Machining time (s)	30,42	20	23,32	20,59
	Minimal Feed rate (mm/min)	650	1000	800	1000
Method	EQUIDISTANT 177	Linear	A-spline	B-spline	C-spline
	Machining time (s)	32,45	23,76	20,31	20,6
	Minimal Feed rate (mm/min)	500	900	1200	1300
Method	EQUIDISTANT 57	Linear	A-spline	B-spline	C-spline
	Machining time (s)	35,29	22,41	18,97	19,97
	Minimal Feed rate (mm/min)	200	60	1250	1100

Ineffective kinetics of linear interpolation described in 1.2.1 caused longer machining time for large number of knots. With lesser knots, linear interpolation deformed the contour which influenced the machining time.

Most time effective for largest number of knots were A-spline and C-spline spline interpolation. For lesser knot, B-spline interpolation resulted in smaller machining time because it wasn't pulled enough towards the knots and created much flatter curve. Real feedrate while applying A-spline interpolation had tendency to decrease down the most out of all spline interpolations especially in local minima. To resolve this problem Sinumerik 840D control system offers short spline blocks compression. It compresses short spline block with greater number of knots and created new set of longer spline blocks. [9].

### 3.14 Discussion

When the imperfection of the measurements is taken in account, a tolerance of mean absolute error inferior to 0,08 mm can be set as a limit for which the employed method interpolates the knots sufficiently. Pairing of methods and distributions that can be considered pass this limit are stated in Table 3.14.1. and Table 3.14.2.

Table 3.14.1 Suitability of studied interpolation methods for equidistant distribution and given input data.

Interpolation method	Distribution	Number of knots	Angle between knots (°)	Mean error (mm)	Suitability
A-spline	Polar	360	1	0,050	OK
B-spline	Polar	360	1	0,099	NOT
C-spline	Polar	360	1	0,049	OK
Linear	Polar	360	1	0,054	OK
A-spline	Polar	72	5	0,041	OK
B-spline	Polar	72	5	0,234	NOT
C-spline	Polar	72	5	0,035	OK
Linear	Polar	72	5	0,088	NOT
A-spline	Polar	24	15	0,676	NOT
B-spline	Polar	24	15	1,241	NOT
C-spline	Polar	24	15	0,382	NOT
Linear	Polar	24	15	0,665	NOT

Table 3.14.2 Suitability of studied interpolation methods for equidistant distribution and given input data.

Interpolation method	Distribution	Number of knots	Distance between knots (mm)	Mean error (mm)	Suitability
A-spline	Equidistant	357	0,38148	0,057	OK
B-spline	Equidistant	357	0,38148	0,047	OK
C-spline	Equidistant	357	0,38148	0,059	OK
Linear	Equidistant	357	0,38148	0,033	OK
A-spline	Equidistant	177	0,76943	0,025	OK
B-spline	Equidistant	177	0,76943	0,055	OK
C-spline	Equidistant	177	0,76943	0,046	OK
Linear	Equidistant	177	0,76943	0,040	OK
A-spline	Equidistant	57	2,38929	0,050	OK
B-spline	Equidistant	57	2,38929	0,330	NOT
C-spline	Equidistant	57	2,38929	0,070	OK
Linear	Equidistant	57	2,38929	0,114	NOT



Equidistant distribution of knots seems more efficient and results in better accuracy even for lesser number of knots. If we compare equidistant distribution with 57 knots and polar distribution with 72 knots the obtained accuracy is comparable even though the number of knot for the equidistant distribution was much higher.

B-spline interpolation was found to be the most inaccurate. The milled curved obtained by B-spline interpolation was deformed much more when given lesser knots for both types distribution of knots.

Linear interpolation results were ameliorated by the fact that 4 out of 11 control points were also knots and linear interpolation passes through the knots (in contrast with B-spline that never does). In reality the facets were present for 72 and less knots for polar distribution and for 57 equidistant knots.

For studied polar distributions A-spline and C-spline were equally performant for 360 knots and starting from 72 knots and less, C-spline interpolated the knots significantly better even though not even C-spline interpolation was sufficient to meet the limit set previously. It's therefore possible to pronounce that the limit for application of C-spline interpolation lays between a knot every  $5^\circ$  and a knot every  $15^\circ$ .

For studied equidistant distributions A-spline and C-spline interpolations were equally performant up until method with 57 knots were A-spline interpolation performed slightly better but that can be caused by an imprecision of the evaluation method described in 3.12.5. For 57 equidistant knots, the distance between two knots is 2,389 mm both A-spline and C-spline interpolate the knots sufficiently. The limit for distance between knots for these two spline interpolation methods is therefore lower than 2,389 mm.

All spline interpolations had shorter machining time compared to linear interpolation, illustrating better kinetics of spline interpolation. While employing the linear interpolation the tool has variable velocity and stops at every knot. It's possible to machine using linear interpolation with the same precision as with spline interpolations when the volume of input data is large enough but with longer machining time.

When we provide enough knots every method of interpolation can be used. For linear interpolation knots placed equidistantly 0,76943 mm apart resulted in sufficient accuracy with mean absolute error inferior to set limit 0,8 mm. For bigger distance between the knots, linear interpolation didn't meet the criteria. B-spline as the second least performant met the limit for the same value of 0,76943 mm between knots but didn't for lesser knots for equidistant neither for polar distribution. A-spline placed second with sufficient accuracy even for only 72 knots spaced by angle  $5^\circ$  using the polar distribution and for equidistant distribution, 2,38929 mm between knots was still sufficient for the A-spline to validate. C-spline validated all employed methods except polar distribution of 24 knots spaced  $15^\circ$  apart. A-spline and C-spline validated for same variations of methods, but C-spline had lower mean absolute error and was therefore concluded to be the most performant out of all used methods.

C-spline interpolation was the most performant even with lesser knots needed. Especial when the knots weren't equidistantly spaced, C-spline wasn't susceptible to differ from the theoretical curve as much as other studied interpolation methods. A-spline was the second most accurate, but manifested some decelerations due to short spline blocks and therefore slightly longer machining time. B-spline was the most likely to deform the curve when given lesser knots out of all studied splines.

Spline interpolations provide better results with lesser knots with the advantage of CNC programs with smaller file size. For example finishing subprogram for equidistant 357 knots for linear interpolation has 40,6 kB while finishing subprogram for equidistant distribution with 177 knots for C-spline has only 20 kB while obtaining similar precision.

## CONCLUSIONS

This study elaborated application of spline interpolations for CNC milling, its advantages and limits. CNC programs employing spline interpolations were designed as a part of this study on a closed symmetrical contour. The programs were then executed and accuracy of spline interpolation methods was examined. Linear interpolation was studied to serve as a comparison for spline interpolations.

Advantages of application of spline interpolations instead of linear interpolation for CNC milling are following:

- smooth toolpath with better surface quality,
- reduction of machining time,
- less input data needed resulting in smaller file size of subprograms containing knot's coordinates.

The comparison of the milled curves using spline interpolation lead to following conclusions:

- C-spline was the most adapted to precisely interpolate a curve even for less input data,
- A-spline placed second when it comes to accuracy of the milled curve,
- NURBS interpolation resulted in deformed curve when provided lesser knots because curve wasn't pulled enough to the knots, the maximal value of weight of the knots programmable in Sinumerik control system had limited value  $PW = 3$ .

In conclusion the limit methods for different spline interpolation are:

- for A-spline: equidistant distribution with 2,38929 mm between knots,
- for B-spline: equidistant distribution with 0,76943 mm between knots,
- for C-spline: equidistant distribution with 2,38929 mm between knots,
- for linear interpolation: equidistant distribution with 0,76943 mm between knots.

Spline interpolation can in some cases replace linear interpolations and bring a significant improvement for machining processes. Spline interpolation has potential to be applied in reverse engineering to machine a new product based on an existing product without the need to create its complex virtual model. Thus method can be employed to machine analytically describable curves without using CAD/CAM systems. Spline interpolations can be applied in reverse engineering or to realize milling of analytically known curves without the need to create a CAD model. To determine limits of spline interpolation in form of maximal distance between knots for every spline interpolation methods, further experimentation is needed with more variations of number of knots per length and distance between knots for equidistant distribution of knots.

**BIBLIOGRAPHY**

1. **BARLIER, C, Ceppetelli, L.** *Méthodes et Production en usinage*. Paris : © Casteilla, 2013. ISBN: 978-2-7135-3546-8.
2. **BAGCI, E.** Reverse engineering applications for recovery of broken or worn parts and re-manufacturing: Three case studies. *Advances in Engineering Software* 40 (2009), 2008, Vols. P.K. 54, 41470, pp. 407–418.
3. **MÉRY, Bernard.** *Machines à commande numérique: de l'étude des structures à la maîtrise du langage*. Paris : Hermès, 1997, 1997. ISBN 28-660-1607-6..
4. **HUMÁR, Anton.** Study material for master's degree studies. *Technologie obrábění – 1. část*. [Online] 2003. [Cited: 18 12 2017.] [http://ust.fme.vutbr.cz/obrabeni/opory-save/TI\\_TO](http://ust.fme.vutbr.cz/obrabeni/opory-save/TI_TO).
5. **CHUDOBA, Milan.** *Základy programování a obsluha CNC strojů: Učební text*. [Online] 2012. [http://www2.sps-jia.cz/~hill/zakl\\_progr.pdf](http://www2.sps-jia.cz/~hill/zakl_progr.pdf).
6. **TOURNIER, Christophe and collective.** *Usinage à grande vitesse*. Paris : Dunod, © 2010. ISBN 978-2-10-051810-4.
7. **VAVRUŠKA, P.** *Postprocesing a výroba tvarově složitých ploch*. [Sborník Konference studentské tvůrčí činnosti - STČ 2009] [ed.] J.MORAVEC. Prague : Fakulta strojní ČVUT v Praze, 2009.
8. *In A new format for 5-axis tool path computation, using Bspline curves.* **LANGERON, J. M. et al.** 2004, Vols. *Computer-Aided Design.* , 36(12), p. 1219-1229.
9. **SIEMENS AG.** *SINUMERIK 840D sl/828 Pro pokročilé*. Nürnberg : Siemens AG, 2010.
10. **ARBENZ, K. WOHLHAUSER, A.** *Analyse numérique*. Laussane : Presses polytechniques romandes, ©1980.
11. **MÜLLER, M., ERDÖS, G., XIROUCHAKIS P.** High accuracy spline interpolation for 5-axis machining. *Computer-Aided design*. pp. 1379-1393, 2004, Vol. 36(13).
12. **ARBENZ, Kurt. BACHMANN, Otto.** *Eléments d'analyse numérique et appliqué*. Laussane : Presses polytechniques romandes, ©1992. ISBN 2-88074-242-0.
13. **VAVRUŠKA, P.** *Machine tool control systems and interpolations of spline type.* **Engineering MECHANICS.** 2012, 19(4), p. 219–229.
14. **BARLIER, C. Ceppetelli, L.** *Méthodes et Production en usinage*. Paris : © Casteilla, 2013. ISBN: 978-2-7135-3546-8.

15. **SIEMENS AG.** *SINUMERIK 840D/840Di/810D, Programming Guide Advanced 03/2004 Edition.* s.l. : © SIEMENS, 2004.
16. **AKIMA, H.** A New Method of Interpolation and Smooth Curve Fitting. *Journal of the ACM (JACM).* Issue 4, October 1970, Vol. 17.
17. **SENCER, Burak.** *Five-axis trajectory generation methods.* Vencouver : The University of British Columbia, Faculty of Graduate Studies (Mechanical Engineering), 2005.
18. **EL BECHIR, M. ZOUBIER, B. MAHER, D. GILLES, D. MOHSEN, A.** Simulation of machining errors of Bspline and Cspline. *The International Journal of Advanced Manufacturing Technology.* 2017, Vols. 89 (n° 9-12), pp. 3323-3330.
19. **CASSELMAN, B.** From Bézier to Bernstein. *American Mathematical Society.* [Online] 2018. [Cited: 02 05 2018.] <http://www.ams.org/publicoutreach/feature-column/fcarc-bezier>.
20. **SIEMENS AG.** *SINUMERIK 840D/840Di/ 810D/FM-NC Short Guide Programming. User Documentation.* 10.2000 . s.l. : © Siemens AG, 1994 - 2000.
21. **SIEMENS AG.** *My SINUMERIK Operate.* s.l. : © Siemens AG, 2013.
22. **HEIDENHAIN.** *Manuel d'utilisation iTNC 530.* Germany : HEIDENHAIN, 2006.
23. **KRAR, S. GILL, A.** *Exploring Advanced Manufacturing Technologies.* New York : Industrial Press Inc., 2003. ISBN 0-8311-3150-0.
24. **STOVER, Christopher, WEISSTEIN, Eric W.** "Polar Coordinates.". [Online] [Cited: 28 04 2018.] <http://mathworld.wolfram.com/PolarCoordinates.html>.
25. UT Calculus. *UT Austin.* [Online] 2009. [Cited: 13 05 2018.] <https://www.ma.utexas.edu/users/m408s/m408d/CurrentWeb/LM10-4-4.php>.
26. **GOLDON, R.** [www.sciencedirect.com](http://www.sciencedirect.com). *Science Direct.* [Online] 21 07 2005. [Cited: 2018 05 14.] <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.413.3008&rep=rep1&type=pdf>.
27. **FAVROLLES, Jean-Pierre.** *Les surfaces complexes: pratique et applications.* Paris : Hermès, 1998. ISBN 28-660-1675-0.
28. **WEISSTEIN, E.W.** Cylindrical Coordinates. *MathWorld - A Wolfram Web Resource.* [Online] 28 04 2018. <http://mathworld.wolfram.com/CylindricalCoordinates.html>.

29. **PRAMET.** E3S N SUMA. *Pramet Ecat*. [Online] 2016. [Cited: 13 05 2018.] <http://ecat.pramet.com/tool.aspx?produkt=03E3S40-09A03%20SUMA&typ=TOOL&lang=Czech>.

30. [www.statisticshowto.com](http://www.statisticshowto.com). *Statistics How To - Mean Error: Definition*. [Online] 25 10 2016. [Cited: 20 05 2018.] <http://www.statisticshowto.com/absolute-error/>.

31. **WANG, Y. et al.** Integration of a 5-axis Spline Interpolation Controller in an Open CNC System. *Chinese Journal of Aeronautics*. 2009, Vols. 22(2), pp. 218-224.

**LIST OF SYMBOLS AND ABBREVIATIONS**

Abbreviation	Unit	Description
2D	[-]	Two dimensional
3D	[-]	Three dimensional
ATC	[-]	Automatic Tool Changer
CAD	[-]	Computer-Aided Design
CAM	[-]	Computer-Aided Manufacturing
CNC	[-]	Computer Numerical Control
ISO	[-]	International Organization of Standardization
MAE	[-]	Mean absolute error
NC	[-]	Numerical Control
NURBS	[-]	Non-uniform rational basis spline
PW	[-]	Parameter of weight (for NURBS interpolation)

Symbol	Unit	Description
A, B, C	[-]	Rotational axes of CNC machine
D	[mm]	Diameter of the tool
G...	[-]	Preparatory functions
F	[-]	Feed rate designation
L	[mm]	
M	[-]	machine reference point
M...	[-]	miscellaneous (auxiliary) functions
N...	[-]	block (line) number
$\rho$	[mm]	rayon of curvature
P	[-]	tool setup point
R	[-]	reference point
$r_t$	[mm]	radius of the tool

$r$	[mm]	distance of a point from the origin of a polar coordinate system
$v_c$	[m.min <sup>-1</sup> ]	cutting speed
$v_f$	[m.min <sup>-1</sup> ]	feed



**LIST OF APPENDICES**

Appendice 1: CD-Rom containing all MATLAB 2018a scripts and NC programs

