

SZABÓ ESZTER
TDK DOLGOZAT

BUDAPESTI MŰSZAKI ÉS GAZDASÁGTUDOMÁNYI EGYETEM
GÉPÉSZMÉRNÖKI KAR
GYÁRTÁSTUDOMÁNY ÉS -TECHNOLÓGIA TANSZÉK



TDK DOLGOZAT

BUDAPESTI MŰSZAKI ÉS GAZDASÁGTUDOMÁNYI EGYETEM
GÉPÉSZMÉRNÖKI KAR
GYÁRTÁSTUDOMÁNY ÉS -TECHNOLÓGIA TANSZÉK

SZABÓ ESZTER

**Bézier görbe alapú trochoidális marási szerszámpályák optimalizálása
gépi tanulás alkalmazásával**

**Bézier curve-based trochoidal milling tool path optimization using
machine learning**

Konzulens:

Dr. Jacsó Ádám

Adjunktus

Budapest, 2024

TARTALOMJEGYZÉK

Előszó.....	v
Jelölések jegyzéke.....	vi
1. Bevezetés	1
1.1. Célkitűzések	1
1.2. Áttekintés.....	1
2. A marási szerszámpályák tervezésének szakirodalmi áttekintése	2
2.1. Nagysebességű megmunkálás	2
2.2. Trochoidális marási stratégia.....	2
2.3. Gépi tanulási módszerek, alkalmazásuk a szerszámpálya-optimalizálásban	4
3. Elvek és módszerek.....	6
3.1. Matematikai háttér.....	6
3.2. A szerszámpályaszakaszok kialakítása	8
3.3. A szerszámpályaszakaszok optimalizálása	10
3.4. Mesterséges neurális hálózat konstrukció	15
3.5. NC program generálás.....	20
3.6. Forgácsolási kísérletek	23
4. Eredmények	26
4.1. Mesterséges neurális hálózati architektúra optimalizálás	26
4.2. Neurális hálózat alapú szerszámpálya generálás szimulációs vizsgálata	28
4.3. Forgácsolási kísérletek	30
4.4. Eredmények értékelése.....	35
5. Összefoglalás	36
6. Summary	37
Felhasznált források	38

ELŐSZÓ

A fogyasztói társadalom piaci igényeinek kielégítése szükségessé teszi a gyártók számára a termelékenység növelését. Ezen cél megvalósításának egyik módja a gyártási folyamatok hatékonyságának növelése. Az elmúlt évtizedben a mesterséges intelligencia fejlesztése nagy teret kapott, így a különböző módszereinek felhasználása is egyre elterjedtebb. A nemlineáris prediktív modellek megfontolatlan használata számos veszélyt rejt, azonban megfelelő keretek között alkalmazva egyes folyamatok időhatékonysága növelhető. A gépi tanulás gyakran alkalmazott módszer a gyártási folyamatok esetében, azonban a szerszámpálya-tervezési célra történő használata kevésbé elterjedt a pályagenerálási feladat komplexitásából fakadóan.

* * *

Köszönöm konzulensemnek, Dr. Jacsó Ádám adjunktusnak a kutatás során, valamint a TDK dolgozatom elkészítésében nyújtott segítséget, iránymutatást, illetve köszönöm Dr. Biró István adjunktusnak a forgácsolási kísérletek során alkalmazott jelfeldolgozó LabView program rendelkezésemre bocsátását.

Budapest, 2024.11.01.

Szabó Eszter

JELÖLÉSEK JEGYZÉKE

<i>Jelölések</i>		
a_{max}	Szerszám gép szánok maximális gyorsulása	$\left[\frac{m}{s^2}\right]$
a_p	Fogásmélység	$[mm]$
b	Horonyszélesség	$[mm]$
d	Szerszámátmérő	$[mm]$
F_r	Eredő erő a kritikus irányban	$[N]$
f_z	Élenkénti előtolás	$[mm]$
MRR	Anyagleválasztási sebesség	$\left[\frac{mm^3}{min}\right]$
r	Szerszámsugár	$[mm]$
rpm	Fordulatszám	$[1/min]$
Q_p	Periódusonként leválasztott anyagmennyiség	$[mm^3]$
s	Ívhossz	$[mm]$
t_p	Periódusidő	$[s]$
v_c	Forgácsolási sebesség	$[m/min]$
v_f	Előtolási sebesség	$[mm/min]$
z	Élek száma	$[db]$
w_{max}	Trochoidális lépésköz	$[mm]$
ρ	Görbületi sugár	$[mm]$
θ	Kontaktszög	$[^\circ]$

Matematikai jelölések

$\mathbf{a}$	A félkövér betű vektort jelöl
$\ \mathbf{a}\ $	Euklideszi norma
$\langle \mathbf{a}, \mathbf{b} \rangle$	Belső szorzat
$\mathbf{r}(t)$	Paraméteres (sík)görbe
$\dot{\mathbf{r}}(t)$	Síkgörbe t paraméter szerinti első deriváltja
$\mathbf{r}'(s)$	Derivált, természetes paraméterezés esetén
$\mathbf{B}(t)$	Bézier függvény
$\mathbf{C}(t)$	Ciklois függvény
$\mathbf{N}(t)$	Normális egységvektor
$\mathbf{T}(t)$	Érintő egységvektor

Rövidítések

ADAM	Adaptive Moment Estimation	Adaptív momentum becslés
ANN	Artificial Neural Network	Mesterséges neurális hálózat
CAM	Computer Aided Manufacturing	Számítógéppel segített gyártás
HSM	High-Speed Machining	Nagysebességű megmunkálás
MRR	Material Removal Rate	Anyagleválasztási sebesség
ReLU	Rectified Linear Unit	Rektifikáló lineáris egység

1. BEVEZETÉS

A hornyok a gépészeti tervezés gyakran előforduló elemei, amiket főként nyomatékviteli feladatok esetén használnak. A piaci igény kielégítésére idő- és költséghatékony gyártási megoldások szükségesek. A hagyományos horonymarás során a ráhagyási alakzat nehéz hozzáférhetősége miatt nagy erők ébrednek, valamint magas szerszámhőmérséklet keletkezik [1], a szerszám kopását, akár törését okozva [2]. Továbbá, a forgács eltávolítása is nehézséget okoz a horony megmunkálása során, ami a forgács újra marását okozhatja, rontva a megmunkált felület minőségét [3].

A trochoidális marási technika alternatívát nyújt az említett problémák elkerülésére horonymegmunkálás esetén is [4]. A trochoidális szerszám pályák egymást bizonyos mértékben átfedő görbeszakaszok láncolatából állnak [5]. A megmunkálás során a szerszám átmérője kisebb a kialakítandó horony szélességénél [6], így a trochoidális marási stratégiával csökkenthető a szerszám radiális bemerülése, azaz a kontaktszög [7]. Ez a tulajdonság a forgácsolás közben keletkező hő hatékonyabb elvezetését eredményezi [8]. Trochoidális marás során a keletkező felület minősége is jobb a hagyományos horonymarással előállított felületekéhez képest [9], valamint a kisebb forgácsvastagság kisebb erők ébredéséhez vezet a szerszámban [10], lehetővé téve az előtolás növelését [11].

1.1. Célkitűzések

A kutatás fő célja olyan szerszám pályá tervező program fejlesztése, amely lehetővé teszi a trochoidális marási stratégia időhatékony alkalmazását horonymaráshoz. A kutatás alapját olyan pályatervezési megoldások képezik, amelyek minimális megmunkálási időt biztosítanak. Ezen módszerek alkalmazása a jelentős számítási igényből kifolyólag nem terjedt el széles körben. A pályatervezés hatékonyságának növelése gépi tanulás segítségével, a program validálása szimulációs és kísérleti úton történik.

1.2. Áttekintés

Az első fejezet tartalmazza a kutatás témájának rövid ismertetését, valamint a célkitűzéseket. A második fejezet az aktuális szakirodalom áttekintését tartalmazza. A harmadik fejezet ismerteti a program megvalósításához szükséges elveket, valamint az felhasznált módszereket és algoritmusokat. A negyedik fejezetben kerülnek bemutatásra a szimulációs, valamint kísérleti eredmények. Az ötödik fejezet a következtetéseket, a munka összefoglalását tartalmazza. A hatodik fejezetben egy rövid, angol nyelvű összefoglaló olvasható.

2. A MARÁSI SZERSZÁMPÁLYÁK TERVEZÉSÉNEK SZAKIRODALMI ÁTTEKINTÉSE

Ebben a fejezetben a nagysebességű megmunkálás irodalmának áttekintésére, majd a trochoidális marási stratégia módszereinek és felhasználásának ismertetésére, végül a gépi tanulási módszer, illetve a szerszám-pálya-tervezésben történő alkalmazásainak bemutatására kerül sor.

2.1. Nagysebességű megmunkálás

A nagysebességű megmunkálás (HSM, High-Speed Machining) alkalmazása esetén a hagyományos technológiához képest jelentősen megnövelt forgácsolási sebességgel dolgozik a szerszám. A korai definíció szerint a HSM olyan megmunkálási technika, amely során a forgácsolási sebesség növelésével csökkennek a forgácsolás közben ébredő erők, valamint a hőképződés [12]. Tlusty szerint a HSM motivációja a fordulatszám növelése, ezzel növelve a percenkénti előtolást és az anyagleválasztási sebességet (MRR, Material Removal Rate) [13]. A HSM előnyei között szerepel a magasabb megmunkálási pontosság, jobb felületi minőség, valamint jobb forgácseltávolítás [14]. A hagyományos szerszám-pálya tervezési módszerek nem ideálisak HSM alkalmazásokra, mivel a sarkos irányváltásoknál a szerszám sebessége csökken, illetve a belső sarkok megmunkálása során ébredő impulzusszerű szerszámterhelés a szerszám töréséhez vezethet [15]. A HSM szerszám-pályák egyik fő kritériuma tehát, hogy nem tartalmazhatnak egy meghatározott görbületi sugárnál kisebb görbületű részeket, így biztosítva az állandó sebességet a megmunkálás során [16]. Követelmény továbbá a G^2 -beli folytonosság [17], azaz a szerszám-pályákat definiáló görbeszakaszok csatlakozási pontjaiban az első és második deriváltak megegyező iránya. Ez a gyakorlatban a sebesség, valamint a gyorsulás vektorok irányának megegyezését, azaz a görbület folytonosságát jelenti. A trochoidális marási szerszám-pályák kielégítik a felsorolt geometriai kritériumokat, emellett trochoidális marás során a forgácsolási erők lefutása jól szabályozott, így a módszer alkalmas HSM alkalmazásra [18].

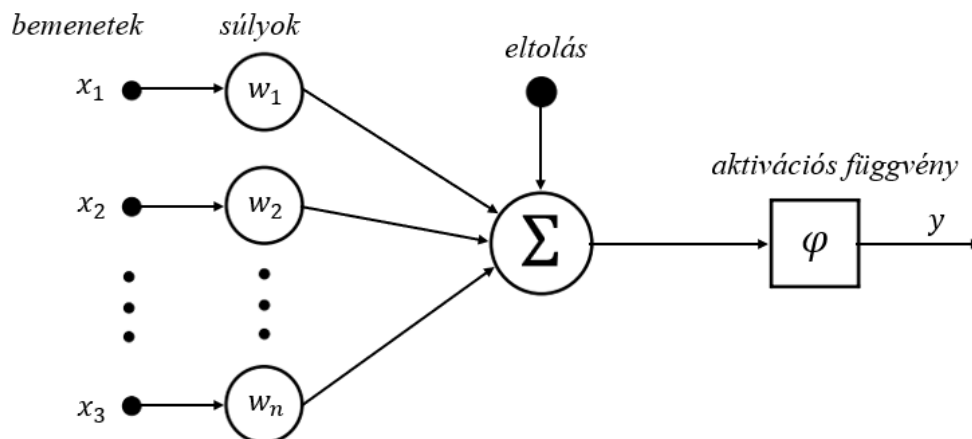
2.2. Trochoidális marási stratégia

Számos különböző megközelítést tanulmányoztak a kutatók a trochoidális marási stratégia alkalmazására. Egyes CAM (Computer Aided Manufacturing - Számítógéppel segített gyártás) rendszerek – többek között a Siemens NX CAM, EdgeCAM, SolidCAM, TopSolid, MasterCAM – alkalmasak trochoidális szerszám-pálya generálásra. A CAM programok rugalmas megoldást

nyújtanak, azonban a kutatások azt mutatják, hogy léteznek hatékonyabb szerszámpálya tervezési módszerek. Wu és társai kör alapú szerszámpályákkal végeztek méréseket, azonban ez a módszer a hagyományos módszereknél kisebb hatékonyságot mutatott [19]. Otkur és társai ciklois-szerű görbeszakaszokat használtak [20]. Megállapították, hogy a javasolt módszer egyszerű felületekhez jól használható, azonban nem alkalmazható olyan felületekhez, amelyek furatokat illetve szigeteket tartalmaznak. Otkur és társai a forgácsolási idő csökkentésére a dupla trochoidális marási technikát is kutatták, melynek során a szerszám bolygómozgásának iránya változik minden periódus végén [20]. Ez a módszer nagyobb forgácsolási erőket okozott, valamint felléptek bizonytalanságok az egyen- és ellenirány váltakozásából fakadóan. Huang és társai ellipszis alapú szerszámpálya-szakaszokat vizsgáltak [21]. A módszerrel a szerszám radiális bemerülése hosszabb szakaszon volt a névleges érték közelében a kör alapú pályákhoz képest, a forgácsolás hatékonysága pedig növekedett a forgácsolási erők egyenletes lefutása mellett. Gu és társai a görbületi sugár alakulását is figyelembe vevő szerszámpálya tervezési algoritmust fejlesztettek ki [22]. Összehasonlították a görbületi variációra szabott szerszámpályákat az ellipszis, a kör és a polinom alapú pályákkal, és méréseik alapján hatékonyabbnak ítélték az új módszert, mivel ebben az esetben a szerszám be- és kilépése gyorsabban valósult meg, így a megmunkálási idő nagyobb hányada zajlott a névleges körülmények között. A felsorolt előnyök mellett azonban a felületi érdesség is növekedett az erők nagyságának változása következtében. Jacsó és társai Bézier görbe alapú módszert használtak szerszámpálya generáláshoz [23]. A görbeszakasz optimalizálását sztochasztikus hegymászó algoritmus segítségével végezték. Forgácsolási kísérleteik azt mutatták, hogy a Bézier görbe alapú módszerrel 40%-kal magasabb volt az MRR a ciklois alapú módszerhez képest, miközben a szerszám terhelése jól szabályozott maradt, továbbá ezzel a módszerrel csökkent a kontaktszög változása. Xu és társai harmadfokú spline alapú módszert alkalmaztak a szerszámpálya forgácsoló szakaszának definiálására [24]. Megmutatták, hogy ez a módszer hatékonyabb, valamint a kontaktszög stabilabb a hagyományos trochoidális marási módszerekhez képest. Jacsó és társai B-spline alapú szerszámpálya tervezési módszert fejlesztettek [17]. Mivel a pályaalak és a pálya hatékonysága közötti kapcsolat összetett, csak szimuláció útján számítható, ezért a szerszámpálya szakaszokat differenciális evolúciós algoritmus segítségével optimalizálták. A kísérletek igazolták, hogy a B-spline alapú trochoidális marási szerszámpályák jobban teljesítenek, mint az NX CAM Adaptive Milling funkciójával tervezett, valamint a ciklois alapú szerszámpályák az MRR szempontjából. Azonban a tanulmány szerint a differenciális evolúciós algoritmusok nagy számítási igénye hátráltatja a gyakorlati alkalmazásokban való elterjedést.

2.3. Gépi tanulási módszerek, alkalmazásuk a szerszámpálya-optimalizálásban

Számos heurisztikus kereső algoritmus létezik, többek között a differenciális evolúciós, genetikus, gradiens, tabu, hegymászó algoritmusok, amelyek képesek végrehajtani különböző optimalizálási feladatokat. A megfelelő algoritmus kiválasztásával ezek a módszerek pontos megoldásokat nyújtanak az adott feladatra. Az elérhető pontosság azonban gyakran magas számítási igénnyel és költséggel jár. A gépi tanulás használható optimalizálási folyamatok gyorsítására a pontosság bizonyos mértékű csökkenése mellett. A gépi tanulás alapja az adathalmaz tulajdonságainak felismerése. A felismerési és információfeldolgozási folyamat hasonlít az agyban lejátszódó folyamatokhoz, azonban a tanulási folyamat az agy természetes folyamataitól számos ponton eltér, például a biológiai esetben folyamatos és párhuzamos a tanulás és információ elérése, míg a gépi tanulás során a két folyamat elkülönülve zajlik. A gépi tanulás során az esetek többségében mesterséges neurális hálózatok (ANN, Artificial Neural Network) hordozzák a megismert információkat [25]. Az emberi agyhoz hasonlóan az ANN is neuronnak nevezett feldolgozó egységekből és az ezek közötti kapcsolatokból áll [26]. Minden neuronhoz tartozik egy aktivációs függvény és egy eltolási érték, valamint minden bemenethez tartozik egy súly (1. Ábra).



1. Ábra. Mesterséges neuron felépítése.

Az ANN architektúra az adathalmaz komplexitása alapján kerül megválasztásra. Az aktivációs függvények az ANN implementálásakor, a neuron definiálásakor kerülnek megválasztásra. Szerepük az ANN komplexitásának növelésében rejlik [27]. A nemlineáris aktivációs függvények értékeitől függően aktiválódik, vagy marad passzív a neuron az adott bemeneti érték hatására, így teszik lehetővé a nem konvex, lineárisan nem szeparálható adathalmazok feldolgozását. A megtanulandó adatokra vonatkozó információk tárolása vektorokban történik. Az adat-vektorok segítségével a

modell neuronjainak súlyai és eltolási értékei optimalizálhatóak a várt kimenet előrejelzése érdekében. A súlyok és eltolási értékek optimalizálására gyakran használt módszer az ADAM (Adaptive Moment Estimation) algoritmus, mivel ez az algoritmus a gradiens módszer segítségével visszavezeti a folyamatot egy egyszerűen implementálható minimalizálási feladatra [28]. A predikciós modell függvények vagy valószínűségi eloszlás formájában tárolható [29]. A betanított ANN használható regresszió, dimenzió csökkentés, sűrűségbecslés, valamint klasszifikáció végrehajtására is [29]. Számos programnyelv, mint például a Wolfram Mathematica, tartalmaz beépített gépi tanulás csomagokat, amelyek segítségével végrehajtható az ANN létrehozása és betanítása. Ezek a csomagok fekete dobozként működnek, azonban minden szükséges információt hordoznak a tanulási folyamatra, valamint a prediktív modellre vonatkozóan. A modellek betanítás után tárolhatóak, így az esetenként hosszas tanítási folyamat elvégzése nem szükséges több alkalommal.

A kutatók a gyártástudományban számos szerszámpálya tervezési feladatra használták a gépi tanulás nyújtotta lehetőségeket. A neurális hálózat alapú megközelítés megfelelőnek bizonyult például az inkrementális lemezalkatrész alakításnál használandó szerszámpálya típus meghatározására, azonban a kimenet visszacsatolása bizonytalanná tette a predikciós modellt, mivel a szerszámkopás is befolyásolta az eredményeket [30]. Balic és társai neurális hálózatokat használtak bonyolult felületek marási stratégiájának megválasztásához a jobb felületi minőség elérése végett [31]. Az ANN segítségével a gyártási folyamat gyorsabbá vált, mivel kevesebb utómunkát volt szükséges. Jacsó és társai az előtolás szabályozására használtak ANN modelleket egyenes és trochoidális maráshoz [32]. A kísérletek alapján az ANN jobban teljesített a hagyományos regressziós technikáknál. A szakirodalom eredményei azt mutatják, hogy a megfelelő ANN modell megválasztásával optimalizálható számos gyártástechnológiai folyamat.

3. ELVEK ÉS MÓDSZEREK

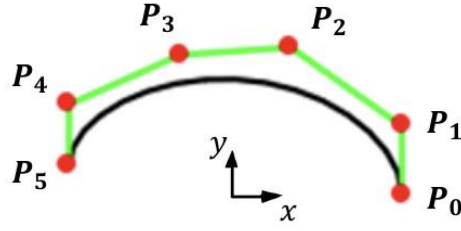
A kutatás témája a Bézier görbe alapú szerszámpálya tervezés, így ebben a fejezetben ismertetésre kerül ennek matematikai háttere. A matematikai összefüggéseket felhasználva a szerszámpályák kialakításának módja, valamint azok optimalizálásának kritériumai kerülnek meghatározásra. Ezen információk felhasználásával megtervezhető és létrehozható a megfelelő neurális hálózat, amely a fő eszköze a kutatás során kifejlesztett szerszámpálya tervezési módszernek. A neurális hálózat segítségével létrehozott Bézier görbeszakaszokból történő NC program generálást végző algoritmus is bemutatásra kerül. A fejezet a kutatási eredmények validálására használt forgácsolási kísérletek körülményeinek ismertetésével zárul.

3.1. Matematikai háttér

A Bézier görbék az approximációs polinomok közé tartoznak. Az ilyen típusú paraméteres függvényekkel definiált sík- vagy térgörbéket a számítógépes modellezésben és grafikában is gyakran használják. A Bézier görbéket kontrollpontok definiálják, amelyeket a görbe megközelít, azonban a kezdő és végpontok kivételével nem halad keresztül azokon. A trochoidális szerszámpályák esetén célszerű az első és az utolsó kontrollpontot úgy megválasztani, hogy a függvény felvegye azokat a kezdő, illetve a végpontjában, mely feltétel a Bézier görbék esetében teljesül. Annak a Bézier görbéhez húzott érintőnek, amelyet az első két kontrollpont, valamint amelyet az utolsó két kontrollpont definiál, a horony széleivel párhuzamosnak kell lennie [23]. Általánosan a Bézier görbéket definiáló egyenlet a következő:

$$B(t) = \sum_{i=0}^n \binom{n}{i} (1-t)^{n-i} t^i \mathbf{P}_i \quad (1)$$

ahol n a Bézier görbe fokát, i az adott kontrollpont indexét, és $\mathbf{P}_i$ az i -edik kontrollpont koordinátáit jelöli. Szerszámpályatervezés során a hat kontrollpontos modellezés esetén elegendő a görbeszakaszok komplexitása az optimalizáláshoz [23]. A szerszámpályákat alakító görbeszakaszok kontrollpontjainak elhelyezése az 2. Ábra szerint történik.



2. Ábra. Bézier görbe kontrollpontjainak elhelyezkedése.

Az kétdimenziós ötödfokú Bézier görbék koordinátáinak értékei adott t pontban az alábbi egyenletek szerint számíthatók:

$$B_x(t) = \sum_{i=0}^5 \binom{5}{i} (1-t)^{5-i} t^i x_i \quad (2)$$

$$B_y(t) = \sum_{i=0}^5 \binom{5}{i} (1-t)^{5-i} t^i y_i \quad (3)$$

A szerszámpályák optimalizálásához szükséges kiszámítani az érintő és a normális egységvektorokat. A síkgörbe érintő egységvektora adott t pontban a következő módon számítható:

$$\mathbf{T}(t) = \frac{\dot{\mathbf{r}}(t)}{\|\dot{\mathbf{r}}(t)\|} \quad (4)$$

ahol

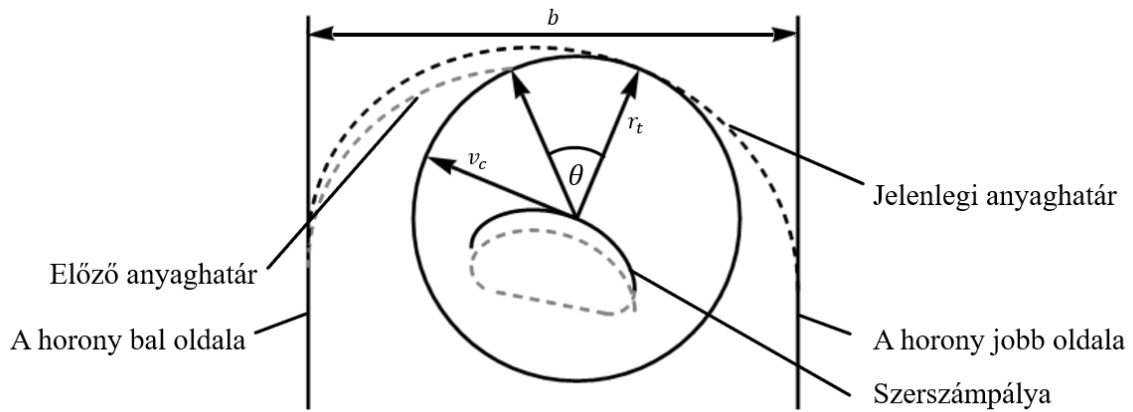
$$\dot{\mathbf{r}}(x(t), y(t)) = \begin{bmatrix} \sum_{i=0}^5 \binom{5}{i} (-(5-i)(1-t)^{5-i-1} \cdot t^i \cdot x_i + i \cdot (1-t)^{5-i} t^{i-1} x_i) \\ \sum_{i=0}^5 \binom{5}{i} (-(5-i)(1-t)^{5-i-1} \cdot t^i \cdot y_i + i \cdot (1-t)^{5-i} t^{i-1} y_i) \end{bmatrix}.$$

A síkgörbék esetén a normális egységvektor adott t pontban az érintő vektor $\pi/2$ vagy $-\pi/2$ radiánnal történő forgatás mátrixával való transzformációjával kapható meg (5).

$$\mathbf{N}(t) = \begin{bmatrix} \cos\left(-\frac{\pi}{2}\right) & -\sin\left(-\frac{\pi}{2}\right) \\ \sin\left(-\frac{\pi}{2}\right) & \cos\left(-\frac{\pi}{2}\right) \end{bmatrix} \cdot \mathbf{T}(t) \quad (5)$$

3.2. A szerszámpályaszakaszok kialakítása

A kontrollpontok koordinátái négy paraméter függvényei; a horonyszélességé, a szerszámtérőé, a kontaktszögé, valamint a minimális görbületi sugaré. A kontaktszög a szerszám keresztmetszetének középpontját a szerszámmal megmunkálandó anyagba történő belépési pontjával összekötő szakasz, valamint a szerszámpálya adott pontjára merőleges vektor által bezárt szög (3. Ábra).



3. Ábra. A kontaktszög (θ) értelmezése.

Az görbeszakasz kontrollpontjainak x koordinátái a horonyszélesség (b), valamint a szerszámsugár (r) függvényeként fejezhetők ki (1. Táblázat). A kontrollpontok elhelyezése az x koordináták szerint egyenközüen történt.

1. Táblázat. A Bézier görbe kontrollpontjainak x koordinátái

A Bézier görbe kontrollpontjainak x koordinátái					
x_0	x_1	x_2	x_3	x_4	x_5
$\left(\frac{b}{2} - r\right)$	$\left(\frac{b}{2} - r\right)$	$\frac{1}{3} \cdot \left(\frac{b}{2} - r\right)$	$-\frac{1}{3} \cdot \left(\frac{b}{2} - r\right)$	$-\left(\frac{b}{2} - r\right)$	$-\left(\frac{b}{2} - r\right)$

A szerszámmal belépési pontja az előző anyaghatár és a szerszám találkozási pontjaként értelmezhető. Innen a kontaktszög (θ) a Bézier görbe érintő vektorának segítségével számítható:

$$\mathbf{Q}(t) = \mathbf{B}(t) + r \cdot \begin{bmatrix} \cos\left(\frac{\pi}{2} - \theta\right) & -\sin\left(\frac{\pi}{2} - \theta\right) \\ \sin\left(\frac{\pi}{2} - \theta\right) & \cos\left(\frac{\pi}{2} - \theta\right) \end{bmatrix} \cdot \mathbf{T}(t) \quad (6)$$

A minimális görbületi sugár – a legkisebb görbület, amelyet a szerszám gép programozott előtolási sebességgel, lassítás nélkül végig tud követni – az alábbiak szerint definiálható:

$$\rho_{min} = \frac{v_c^2}{a_{max}} \quad (7)$$

ahol v_c a forgácsolási sebesség, és a_{max} a szervomotorok maximális gyorsulása [23]. A periódusidő – a szerszám pálya egy periódusának bejárásához szükséges idő – kiszámításához szükséges az ívhossz ismerete. A paraméteres síkgörbék ívhossza adott t_0 és t_1 pontok között az alábbi képlettel határozható meg:

$$s = \int_{t_0}^{t_1} \sqrt{\left(\frac{dx(t)}{dt}\right)^2 + \left(\frac{dy(t)}{dt}\right)^2} dt \quad (8)$$

Az ötödfokú Bézier görbék esetében nem létezik zárt megoldása az ívhossz integrálnak, így az ívhossz a síkgörbe megfelelő finomságú egyenes szakaszos közelítésével, az egyes szakaszok hosszának összegzésével számítható. Az ívhossz értékének ismeretében kiszámítható a periódusidő:

$$t_p = \frac{s}{v_f} \quad (9)$$

A görbületi sugár kiszámításához célszerű áttérni természetes paraméterezésre. A paraméter kiszámításához szükséges az ívhossz meghatározása t függvényében:

$$s = \int_0^t \sqrt{\left(\frac{dx(\tau)}{d\tau}\right)^2 + \left(\frac{dy(\tau)}{d\tau}\right)^2} d\tau \quad (10)$$

A természetes paraméter a (10)-es egyenlet t -re rendezésével kapható meg. A természetes paraméterrel megadott görbealak numerikus megfelelője az egyenlő hosszú egyenes szakaszokkal történő közelítés.

Az adott pontbeli görbületi sugár természetes paraméterezés esetén a következő analitikus, valamint közelítő numerikus egyenlet szerint számítható:

$$\rho(s) = \frac{\mathbf{T}(s)'}{|\mathbf{T}(s)'|} = \frac{\|\mathbf{B}(s) - \mathbf{B}(s + ds)\|}{\arccos\left(\frac{\langle \mathbf{T}(s), \mathbf{T}(s + ds) \rangle}{\|\mathbf{T}(s)\| \|\mathbf{T}(s + ds)\|}\right)} \quad (11)$$

ahol s az ívhossz szerinti paraméter, ds egy megfelelően kis szám.

A megmunkált geometria adott periódusbeli határát a Bézier görbe evolútájának $\mathbf{N}(t)$ irányában $(\rho - r)$ -rel való eltoltja adja meg:

$$\mathbf{e}(t) = \mathbf{B}(t) + r \cdot \mathbf{N}(t) \quad (12)$$

A maximális trochoidális lépésköz (w_{max}) a $\theta = \theta_{max}$ -hoz tartozó, a 6. egyenlet által definiált $Q(t)$ görbe, valamint az aktuális anyaghatár y irányú távolságának a görbeszakasz mentén vett minimuma. A periódusonként eltávolított anyagtérfogat a w_{max} érték segítségével számítható ki:

$$Q_p = b \cdot a_p \cdot w_{max} \quad (13)$$

Utolsó lépésként az MRR kiszámítása történik (14).

$$MRR = \frac{Q_p}{t_p} = \frac{b \cdot a_p \cdot w}{s/v_f} \quad (14)$$

A szerszámpályaszakaszok kontrollpontjai egységnyi sugarú szerszámmra vannak optimalizálva, mivel az 1 mm-es szerszámsugár egyszerűsíti a számításokat. A kontrollpontok 2 mm-től eltérő szerszámtérő esetén az optimalizált pontok szerszámsugárral való léptékezésével kaphatók meg.

3.3. A szerszámpályaszakaszok optimalizálása

A célkitűzések között szerepel a megmunkálás hatékonyságának növelése, ennek megfelelően az optimalizálás célfüggvényeként az MRR választható. Az optimalizálás során továbbá szükséges figyelembe venni a geometriai korlátokat. A maximális kontaktszögre vonatkozó feltétel a szerszámterhelés szabályozását biztosítja, a minimális görbületi sugár pedig a megvalósíthatóságot. A feltételeket figyelembe véve az MRR az 1. Algoritmus szerint számítható.

1. Algoritmus

function fitness_szamitasa(pontok, b, θ , ρ , a_p, t_p, v_f)

comment Ez a függvény kiszámítja az MRR értéket

Bemenet: pontok, horonyszélesség (b), kontaktszög (θ), minimális görbületi sugár (ρ), fogásmélység (a_p), periódusidő (t_p), előtolási sebesség (v_f).

Kimenet: MRR.

function Bezier(pontok, s)

Az s paraméterhez tartozó, pont array által definiált Bézier függvény koordinátáinak kiszámítása az (2), (3) egyenlet szerint, természetes paraméterezés.

return {B_x, B_y};

end

function erinto(gorbe, s)

A paraméteres görbe érintő vektorának kiszámítása és normálása az s kiértékelési pontban a (4) egyenlet szerint.

return {erinto_x, erinto_y};

end

function normal(erinto_x, erinto_y, t)

Érintő vektor forgatása -90 fokkal a (5) egyenlet szerint.

return {normal_x, normal_y};

end

function belepesi_pont(s)

Szerszámbelépési pont kiszámítása a (6) egyenlet szerint.

return {belepes_x, belepes_y};

end

function gorbuleti_sugar(s)

Az s pontbeli görbületi sugár számítása a (11) egyenlet szerint.

return { ρ };

end

function trochoidalis_lepeskoz(pontok)

A maximális trochoidális lépésköz számítása.

return {w_max};

end

function ivhossz(pontok)

Az ívhossz integrál kiszámítása a (8) egyenlet szerint.

return {s};

end

real w_max = trochoidalis_kiosztas(pontok);

real s = ivhossz(pontok);

real t_p = s/v_f;

real MRR = (w_max * a_p * b)/t_p;

return {MRR};

end

A Jacsó és társai [23] számításait alapul vevő iteratív algoritmus segítségével történt a kiindulási Bézier görbe generálás. A kiindulási y koordináták meghatározásához szükséges a ciklois alapú szerszámpálya generálás. A ciklois-szerű szerszámpályák paraméteres egyenlete a következő:

$$C(t) = \left[\left(\frac{b}{2} - r \right) \cdot \cos(t), \left(\frac{b}{2} - r \right) \cdot \sin(t) + w_0 \cdot \frac{t}{2\pi} \right] \quad (15)$$

ahol w_0 a ciklois szerszámpályáknál alkalmazott lépésköz. Az további y koordináták meghatározása a 2. Algoritmus segítségével történik.

2. Algoritmus

function uj_pont ($y_0, y_1, y_2, y_3, y_4, y_5, \delta$)

comment Az uj_pont függvény generálja a Bézier görbe y koordinátáinak új értékét.

Bemenet: eredeti y koordináták ($y_0, y_1, y_2, y_3, y_4, y_5$) és maximális lépés nagyság (δ).

Kimenet: új y koordináták ($y_{uj}[0], y_{uj}[1], y_{uj}[2], y_{uj}[3], y_{uj}[4], y_{uj}[5]$).

array $y[6] = \{y_0, y_1, y_2, y_3, y_4, y_5\}$;

$y_{uj}[0] = y_0$;

do

begin

for $j := 1$, **step** 1 **until** 6 **do**

begin

real tmp = (random szám - δ és δ);

$y_{uj}[j] = y[j] + \text{tmp}$;

end

while az $y_{uj}[1]$ és $y_{uj}[2]$, valamint $y_{uj}[4]$ és $y_{uj}[5]$ közötti vektorok nem megfelelő irányban állnak

end

return $\{y_{uj}[0], y_{uj}[1], y_{uj}[2], y_{uj}[3], y_{uj}[4], y_{uj}[5]\}$;

end

A kereső algoritmus az eredeti pontokból kiindulva a 2. Algoritmust felhasználva keresett optimális görbeszakaszt a szerszámpályához (3. Algoritmus).

3. Algoritmus

```
function optimalis_palya(x1, x2, x3, x4, b,  $\theta$ ,  $\rho_{\min}$ )  
comment      Ez a függvény optimalizálja a szerszámpálya szegmenseket.  
               Bemenet: kontrollpontok x koordinátái, horonyszélesség, kontaktszög, minimális  
               görbületi sugár.  
               Kimenet: kontrollpontok y koordinátái.  
array ciklois0[2] = (ciklois x és y koordinátáinak kiszámítása  $t = 0$  pontban a (15) egyenlet szerint);  
array cikloispi[2] = (ciklois x és y koordinátáinak kiszámítása  $t = \pi$  pontban a (15) egyenlet szerint);  
array pontok[6][2] = {ciklois0, {x1, y1init}, {x2, y2init}, {x3, y3init}, {x4, y4init}, cikloispi};  
fitness = fitness_szamitasa(pontok, b,  $\theta$ ,  $\rho$ );  
 $\delta_0 = 0.125$ ;  $\delta_{\min} = 0.001$ ; i = 1; imax = 1000; jmax = 100;  
for  $\delta = \delta_0$ , let  $\delta = \delta/5$  until ( $\delta \geq \delta_{\min} \wedge i \leq \text{imax}$ ) do  
  begin  
    while  $i \leq \text{imax}$ , step 1, do  
      begin  
        for j=1, step 1 until  $j \leq \text{jmax}$ , do  
          begin  
            tesztpontok = uj_pont(pontok,  $\delta$ );  
            tesztfitness = fitness_szamitasa(tesztpontok, b,  $\theta$ ,  $\rho$ )  
            if  $\rho \leq \rho_{\min}$  then  
              begin  
                continue;  
              end  
            if tesztfitness[0] > fitness[0] then  
              begin  
                break;  
              end  
            end  
          end  
        pontok = tesztpontok;  
        fitness[0] = tesztfitness[0];  
      end  
    end
```

end

return{pontok};

A sztochasztikus kereső algoritmus 44 különböző bemeneti kombinációhoz generált optimális szerszámpályát. A görbét meghatározó paraméterek és a görbe y koordinátái bemeneti és kimeneti mátrixokat alkottak. A bemeneti mátrix a horonyszélesség és a szerszámtérő hányadosát, a minimális görbületi sugarat, valamint a maximális kontaktszöget tartalmazta minden görbéhez, a kimeneti mátrix pedig a Bézier görbék y koordinátaiból állt.

A bemeneti paraméterek lehetséges értékeit a 2. Táblázat tartalmazza. Egyes bemeneti paraméterkombinációkhoz nem talált a kereső algoritmus optimális Bézier görbeszakaszt a meghatározott maximális iteráció alatt, mivel a kisebb horonyszélességek esetén a nagy görbületi sugarak nem megvalósíthatóak. A lehetséges 64 kombinációból 44 esetben lehetett csak optimális Bézier görbeszakaszt definiálni. Ez a 44 eset jelentette azt az adathalmazt, amelyen történt a gépi tanulás megvalósítása.

2. Táblázat. A bemenetek lehetséges értékei.

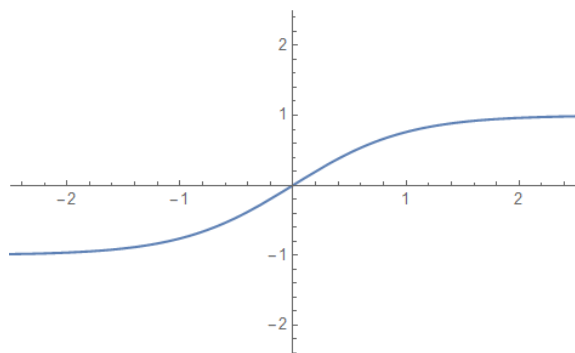
Paraméter		Lehetséges értékek			
b/d	[mm]	1,25	1,5	1,75	2
θ_{max}	[°]	15	30	45	60
ρ_{min}	[mm]	0,1	0,2	0,3	0,4

3.4. Mesterséges neurális hálózat konstrukció

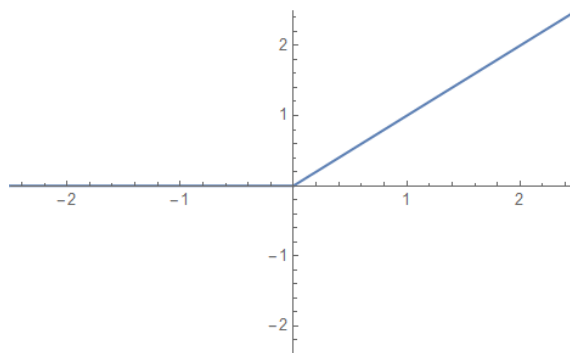
A Bézier görbe optimalizálási folyamat gyorsítása egy Wolfram Mathematica 12-ben létrehozott ANN segítségével történt. A bemeneti és kimeneti vektor párok (a bemeneti és kimeneti mátrixok összetartozó sorai) tanító és validációs adathalmazokat alkottak. A kettéválasztás elve kísérleti információkon alapult. A tanító adathalmaz a vektorpárok felét tartalmazta, a validációs halmaz pedig a maradékot. A vektorpárok egyenletesen oszlottak meg a lehetséges bemeneti adatok (2. Táblázat) alapján.

Egy optimalizáló program készült a legalkalmasabb ANN architektúra megtalálására, mivel a megfelelő neuronszám megválasztására nem létezik általános algoritmus. A program különböző modelleket készített az ANN teljesen kapcsolt rétegeinek számának, valamint a rétegenkénti

neuronok számának változtatásával. Az aktivációs függvények a hiperbolikus tangens (4. a) Ábra), illetve a ReLU (4. b) Ábra) függvények voltak.



4. Ábra. a) Hiperbolikus tangens függvény.



b) ReLU függvény.

A túlillesztés elkerülése érdekében a tanítási folyamat megállt, amennyiben a veszteség függvény nem javult az előző körökben (4. Algoritmus).

4. Algoritmus

function tanitas(neuralis_halozat)

comment Ez a függvény hajtja végre a neurális hálózat tanítási folyamatát.

Bemenet: neurális hálózat

Kimenet: betanított neurális hálózat.

Kezdeti iteráció végrehajtása

array veszteseg_ertekek = [];

Veszteség kiszámítása (átlagos négyzetes hiba az előrejelzett és a validációs kimenetek közötti különbségből számítva)

Append veszteseg_ertekek **with** számított veszteség;

integer turelem = (Egész szám 200 és 500 között, tapasztalati alapon kiválasztva az egyes ANN típusokhoz);

integer j = 0;

do

begin

increment j **by** 1;

Következő iterációs kör végrehajtása;

Veszteség számítása;

Append veszteseg_ertekek **with** veszteség;

if loss_values[j]>=loss_values[j-1] **then**

turelem csökkentése 1-gyel;

end

while turelem>0

end

Az ANN-ek osztályozása az MRR értékek, valamint a minimális görbületi sugárra vonatkozó korlátozó feltételnek való megfelelés alapján történt. Mivel a minimális görbületi sugár biztosítja az állandó és ideális forgácsolási sebességgel történő megmunkálás lehetőségét, így annak megfelelése jelentette az első számú feltételt. A minimális görbületi sugárra ANN architektúránként három mérőszám vonatkozott. Kiszámításra került, hogy a 22 validációs adat közül mennyi esetben volt a megengedettnél kisebb a minimális görbületi sugár, valamint ezen eltérések

számértékének összege, továbbá, hogy mennyi esetben volt a megengedett görbületi sugár 10%-ánál nagyobb az eltérés.

Az optimális ANN feladata olyan szerszámpályák generálása, amelyek a tanító adatokéhoz közel eső MRR értékeket biztosítanak, így ez volt a második számú feltétel. Az MRR megfeleléségre két mérőszám vonatkozott, az első azoknak az eseteknek a számát adta meg, amelyekben az ANN által generált szerszámpálya MRR értéke, valamint a validációs adathoz tartozó MRR érték közötti eltérés nagyobb a validációs MRR 10%-ánál. A másik mérőszám az összes validációs adathoz számított MRR különbségek mediánja volt. Az optimalizálási folyamatot az 5. Algoritmus hajtotta végre.

5. Algoritmus

function halozat_optimalizalas(tanito_adatok, validacios_i, validacios_o)

comment A program létrehozza a különböző ANN architektúrákat, végrehajtja a tanítási folyamatot és kiértékeli a validációs adathalmaz fitness értékeit.

Bemenet: bemeneti-kimeneti párok (tanito_adatok) alkotta array, b , θ és ρ értékek szerint sorbarendezve, valamint a validációs adathalmaz bemeneti vektora, és kimeneti vektora.

Kimenet: Az egyes ANN-ek MRR értékei.

function neuron_szam(n) = *(a neuronok számának meghatározása az n-edik rétegben, tesztelt függvények: $n, 3*n, 5*n, 3*5*2^{(n-1)}$);*

end

for j= *rétegek minimális száma*, **step** 1 **until** j= *rétegek maximális száma* **do**

begin

create halozat, *egyrétegű neurális hálózat, bemenetek száma:3, kimenetek*

száma: neuron_szam[0];

for k =1, **step** 1 **until** k=j **do**

begin

Append halozat **with** *egy réteg, bemenetek száma: neuron_szam[k-1], kimenetek száma: neuron_szam[k];*

end

Append halozat **with** *egy réteg, bemenetek száma: neuron_szam[j], kimenetek száma: 5;*

 betanított=tanitas(halozat);

 halozat *kimeneteinek számítása a validacios_i bemenetekhez;*

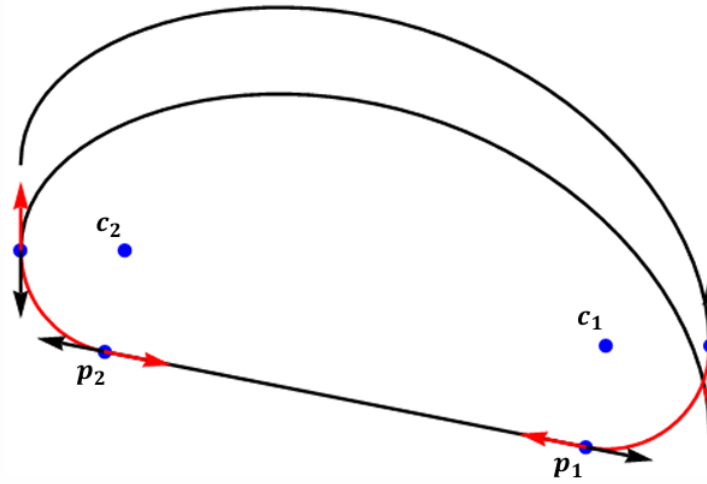
MRR és ρ_{min} számítása a hálózat kimenetekhez, összehasonlítás a validacios_o-hoz számított értékekkel;

 return{j,MRR, ρ_{min} }

end

3.5. NC program generálás

A Bézier görbéket, a számított trochoidális lépésközt, valamint a minimális görbületi sugarat alapul véve készültek a szerszámpályák egy Wolfram Mathematica 12-ben készített program segítségével. A szerszámpálya forgácsoló szegmenseit definiálták a Bézier görbék. Az összekötő szakaszokat a minimális görbületi sugárral megegyező rádiuszú körívek, valamint egy ezeket összekötő egyenes szakasz alkotta. Minden egymást követő görbeszakasz érintőlegesen kapcsolódott (5. Ábra).



5. Ábra. A szerszámpálya szegmensek tangenciális csatlakozása.

Mivel a Bézier görbék kezdő- és végpontjaihoz húzott érintők az y tengellyel megegyező irányúak, ezért az érintőkörök középpontjainak y koordinátái megegyeznek a görbe ezen pontokbeli y koordinátaival. Az érintőkörök középpontjainak x koordinátái megkaphatók a görbe x koordinátáinak ρ_{min} -el való növelésével, illetve csökkentésével. Jelölje $\mathbf{c}_1$ a w_{max} -szal y irányba eltoltt Bézier görbeszakasz kezdőpontjához rajzolható érintőkör középpontját, valamint $\mathbf{c}_2$ az eredeti Bézier görbeszakasz végpontjába rajzolható érintőkör középpontját. Jelölje $\mathbf{p}_1$ és $\mathbf{p}_2$ a köröket összekötő egyenes szakasz végpontjainak koordinátáit. Ekkor a következő egyenletrendszer megoldásával megkapható $\mathbf{p}_1$ és $\mathbf{p}_2$:

$$\left\{ \begin{array}{l} \langle \mathbf{p}_1 - \mathbf{c}_1, \mathbf{p}_1 - \mathbf{c}_1 \rangle = \rho_{min}^2 \\ \langle \mathbf{p}_2 - \mathbf{c}_2, \mathbf{p}_2 - \mathbf{c}_2 \rangle = \rho_{min}^2 \\ \langle \mathbf{p}_1 - \mathbf{p}_2, \mathbf{p}_1 - \mathbf{c}_1 \rangle = 0 \\ \langle \mathbf{p}_1 - \mathbf{p}_2, \mathbf{p}_2 - \mathbf{c}_2 \rangle = 0 \\ p_{1,y} \leq c_{1,y} \\ p_{2,y} \leq c_{2,y} \end{array} \right. \quad (16)$$

Egy szerszámpályaszakasz meghatározása után a program diszkretizálta a forgácsoló, illetve összekötő szakaszokat. A kapott pontmátrix segítségével történt az NC kód generálása. A szerszámpálya kizárólag G1-es szakaszokat tartalmazott (6. Algoritmus).

6. Algoritmus

function palyageneralas(i,o,r,n)

comment Ez a függvény generálja az n periódusból álló szerszámpályát.

Bemenet: horonyszélességből, kontaktszögéből, és minimális görbületi sugárból álló array (i), kimeneti array (o), szerszámsugár (r), periódusok száma (n).

Kimenet: NC program txt formátumban.

pontok = *(x és y koordinátákból álló array, a koordináták a szerszámsugárral (r) felszorozva);*

$\rho = i[3]*r;$

wmax = *(wmax számítása, mint az 1. Algoritmusban);*

s = *(Ívhossz számítása, mint az 1. Algoritmusban);*

comment A forgácsoló szegmenst definiáló szakaszok számítása

gorbereszek = *(s/0.15 lefelé kerekítve);*

dp = 1/gorbereszek;

diszkret_fuggvény = *(A pontok array által definiált Bézier görbe diszkretizálása a $t = [0,1]$ intervallumon, dp növekménnyel);*

eltolas = max(diszkret_fuggvény)[2]+r;

comment Összekapcsoló szakaszok számítása

c1 = *(A wmax-szal eltolt Bézier görbe szakasz érintőkörének középpontjának koordinátái a $t=0$ pontban);*

c2 = *(Az eredeti Bézier görbe szakasz középpontjának koordinátái a $t=1$ pontban);*

p1,p2 = *(a (16) egyenlet szerint számított érintő kezdő és végpontjainak koordinátái);*

diszkrét_iv1 = *(a p1 pontot a wmax-szal eltolt Bézier görbe kezdőpontjával összekötő ív diszkretizálása);*

diszkret_iv2 = *(az eredeti Bézier görbe végpontját és a p2 pontot összekötő ív diszkretizálása);*

comment NC kód generálása

stream = *(fájl elérési útvonalának definiálása);*

```

write technológiai paraméterek to stream
for j=0, step 1 until j<n do
  begin
    for k=0, step 1 until length(diszkret_fuggveny) do
      begin
        write "G1 X", diszkret_fuggveny[k][0], " Y", diszkret_fuggveny[k][1]-eltolas
        +k*wmax;
        write end line;
      end
    for k = 0, step 1 until length(diszkret_iv2) do
      begin
        write "G1 X", diszkret_iv2 [k][0], " Y", diszkret_iv2 [k][1]-eltolas+k*wmax;
        write end line;
      end
    write "G1 X", p1[0], " Y", p1[1]-eltolas+k*wmax;
    write end line;
    for k = 0, step 1 until length(diszkret_iv1) do
      begin
        write "G1 X", diszkret_iv1 [k][0], " Y", diszkret_iv1 [k][1]-eltolas+k*wmax;
        write end line;
      end
    end
  end
write szerszám pálya záró paraméterek

```

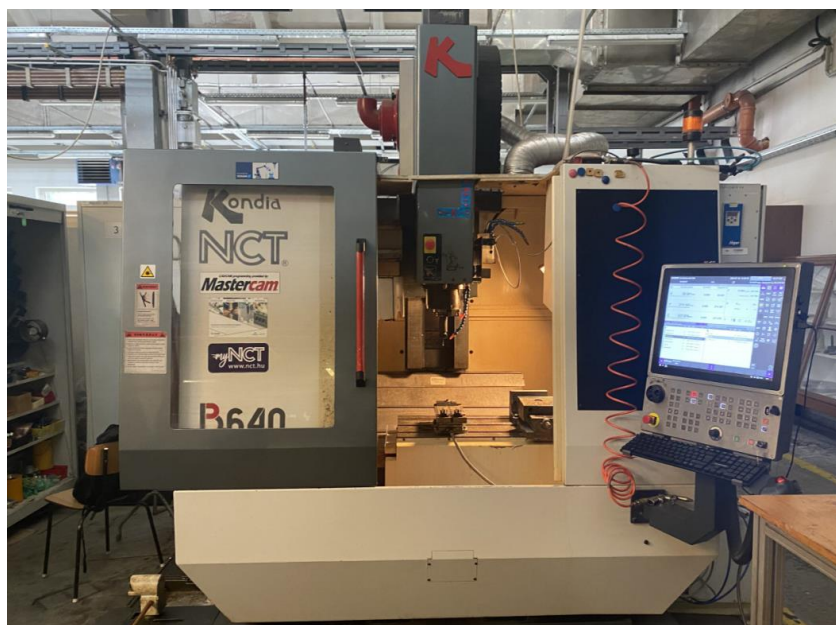
Öt szerszám pálya készült az ANN által generált Bézier görbék segítségével 10 mm-es szerszámátmérőhöz, valamint egy szerszám pálya készült a kereső algoritmus által optimalizált Bézier görbékkel (3. Táblázat). Összehasonlítási alapként további három szerszám pálya készült a Siemens NX CAM Adaptive Milling segítségével.

3. Táblázat. A generált szerszámpályák bemeneti paramétereit.

Sorszám	Pályagenerálási módszer	b [mm]	θ_{max} [°]	ρ_{min} [mm]
1.	Kereső algoritmus	15	45	0,75
2.	ANN	15	45	0,75
3.	ANN	12,5	45	0,75
4.	ANN	17,5	60	0,75
5.	ANN	15	30	0,75
6.	ANN	15	60	0,75
7.	NX CAM	15	30	-
8.	NX CAM	15	45	-
9.	NX CAM	15	60	-

3.6. Forgácsolási kísérletek

Az erőmérési kísérlet egy Kondia B-640 háromtengelyes megmunkálóközpont segítségével történt (6. a) Ábra). A szerszám egy 10 mm-es, 3 élű Smicut M1010C25_LC szármaró volt (6. b) Ábra).



6. Ábra.

a) Kondia szerszámgép.



b) Smicut szármaró.

Az alkalmazott technológiai paramétereket a 4. Táblázat tartalmazza.

4. Táblázat. Forgácsolási paraméterek.

Forgácsolási paraméterek			
Forgácsolási sebesség	v_c	200	[m/min]
Élenkénti előtolás	f_z	0,05	[mm]
Élek száma	z	3	[db]
Fordulatszám	rpm	6366,198	[1/min]
Előtolási sebesség	v_f	954,930	[mm/min]
Fogásmélység	a_p	2	[mm]

A forgácsolás során ébredő erők mérése egy Kistler 9257B piezoelektromos erőmérő szenzor segítségével történt (7. Ábra).



7. Ábra. Kistler 9257B piezoelektromos erőmérő szenzor.

Az erőmérő hosszanti iránya megegyezett az abszolút koordináta-rendszer y tengelyének irányával. Ennek beállítása mérőórával történt. A munkadarab téglatest alakú volt, anyaga Al6082 alumínium ötvözet, méretei 100 x 175 x 120 [mm]. Az axiális fogásmélység egyenletességének biztosítása síkmarással történt. A kísérlet során az erőmérő jele egy Kistler Multichannel Charge Amplifier Type 5070 erősítőn (8. Ábra), majd egy National Instruments USB 4431 analóg-digitális jelátalakítón haladt át. A digitális jel feldolgozása egy LabView környezetben megírt mérőszoftver segítségével történt (8. Ábra), amelyet Dr. Biró István adjunktus bocsátott rendelkezésemre.



8. Ábra. Jelerősítő (bal oldalon), jelátalakító (középen), jelfeldolgozó szoftver (jobb oldalon).

A jelfeldolgozás kimenete Technical Data Management Streaming (TMDS) formátumú fájl volt, ami egy közvetlen bináris kód [33]. Ez a formátum biztosította a gyors és pontos jelfeldolgozást [34]. A jelfeldolgozó szoftver 2000 adategységből álló csomagokat töltött a memóriába. A mintavételezési frekvencia 12 kHz, így a jelen mérések során szerszámfordulatonként 113 mérés történt. A mérési pontok gyakorisága miatt analógnak tekinthető a mérés. A mérések elvégzését követte a TMDS formátum txt formátummá alakítása, majd a Wolfram Mathematica 12 segítségével végzett adatfeldolgozás. A kritikus erők x és y irányban mérhetők. Az eredő erő kiszámítása a mérési adatok felhasználásával az alábbi módon történt:

$$F_r = \sqrt{F_x^2 + F_y^2} \quad (17)$$

ahol F_x az x irányú, és F_y az y irányú erő.

4. EREDMÉNYEK

Ebben a fejezetben a kifejlesztett program alapját képző neurális hálózat, valamint a szerszámpályákon végzett szimulációk és mérések eredményei kerülnek ismertetésre. A fejezet az eredmények értékelésével zárul.

4.1. *Mesterséges neurális hálózati architektúra optimalizálás*

A neurális hálózatok tesztelése a 3.3. fejezetben meghatározott adathalmazok segítségével történt. Az ismertetésre kerülő statisztikai adatok (5. Táblázat) a 22 adatpárból álló validációs halmazra vonatkoznak. Az alábbi meghatározások az 5. Táblázat megértését szolgálják:

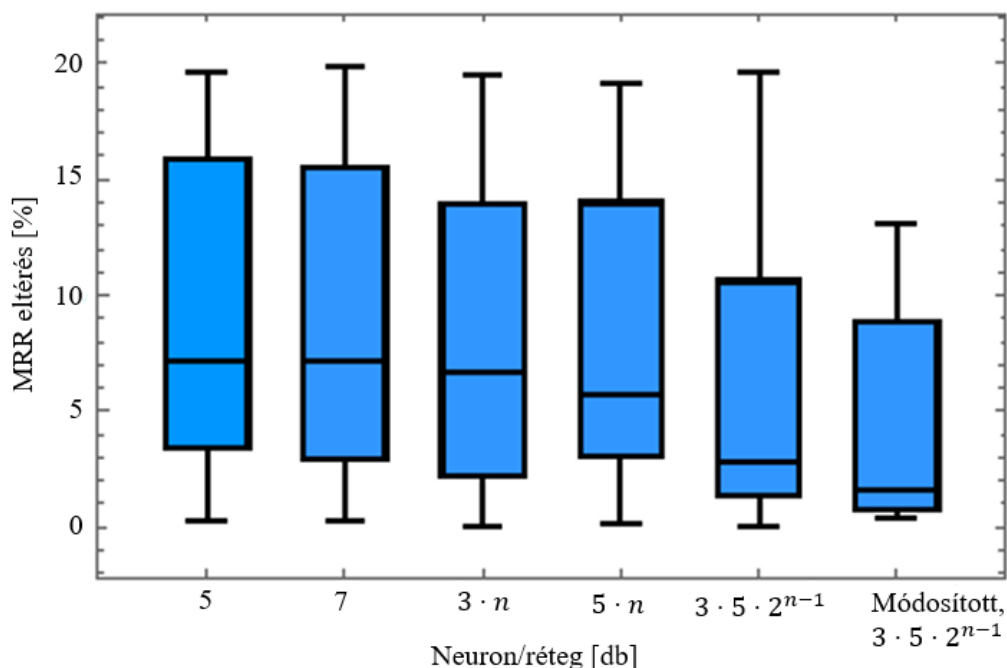
- A) Azon bemeneti kombinációk százalékos aránya, amelyek esetén az ANN által generált szerszámpálya legkisebb görbületi sugara az elvárt minimális görbületi sugárnál kisebb.
- B) Azon bemeneti kombinációk százalékos aránya, amelyek esetén az ANN által generált szerszámpálya legkisebb görbületi sugara kisebb, mint az elvárt érték és az eltérés a több, mint 10%-a a minimális görbületi sugárnak.
- C) Az A-esetbeli szerszámpályák legkisebb görbületi sugarának az elvárt minimális görbületi sugártól való eltérésének számszerű összege.
- D) Azon bemeneti kombinációk százalékos aránya, amelyek esetén az ANN által generált kimenet, valamint a kereső algoritmus által generált kimenet MRR értékei közötti eltérés több, mint 10%-a a kereső algoritmus által generált kimenet MRR értékének.
- E) Az ANN által generált kimenet, valamint a kereső algoritmus által generált kimenet MRR értékei közötti eltérés mediánja százalékban kifejezve.

5. Táblázat. A vizsgált neurális hálózat családok optimális egyedei.

ANN család a neuronok rétegenkénti száma szerint ¹	Rétegek száma	A) [%]	B) [%]	C) [mm]	D) [%]	E) [%]
Konstans, 5	4	27,27	9,09	0,098	9,09	7,177
Konstans, 7	17	27,27	9,09	0,098	9,09	7,171
$3 \cdot n$	7	27,27	9,09	0,098	9,09	7,652
$5 \cdot n$	17	31,81	18,18	0,108	0	5,700
$3 \cdot 5 \cdot 2^{n-1}$	8	22,72	18,18	0,059	18,18	2,766
$3 \cdot 5 \cdot 2^{n-1}$ (módosított)	7	22,72	9,09	0,067	0	1,620

¹ n az adott réteg sorszáma, ahol a számozás az első rejtett rétegtől indul, 1-gyel kezdve.

Az ANN komplexitása növelhető az aktivációs függvények változtatásával, ezáltal nagyobb pontosság érhető el. Az rétegek aktivációs függvényeinek meghatározása a hiperbolikus tangens, valamint a ReLU függvények kombinációinak szisztematikus tesztelésével történt. A $3 \cdot 5 \cdot 2^{n-1}$ típusú ANN esetében az aktivációs függvény kombináció módosításával a rétegszám csökkentése mellett jobb prediktív modell jött létre. A statisztikai elemzés alapján a görbületi sugárbeli eltéréseket tekintve a módosított $3 \cdot 5 \cdot 2^{n-1}$ családba tartozó ANN bizonyul a legalkalmasabbnak a Bézier görbe optimalizálási feladatra. Az egyes validációs adatokhoz számított MRR-beli eltérések mediánja alapján is ez tekinthető az ideális ANN konstrukciónak. Az MRR értékbeli kiugró eltérések is a módosított $3 \cdot 5 \cdot 2^{n-1}$ családba tartozó ANN esetében a legkisebbek, valamint az eltérések statisztikai eloszlása is kedvező (9. Ábra).



9. Ábra. A százalékos MRR eltérések eloszlása a validációs adatokon a kereső algoritmussal generált szerszámpályákhoz viszonyítva.

A módosított $3 \cdot 5 \cdot 2^{n-1}$ családba tartozó ANN rétegenkénti neuron számát, valamint az aktivációs függvényeket a 6. Táblázat tartalmazza.

6. Táblázat. Optimális neurális hálózat.

Optimalizált neurális hálózat								
Réteg (n)	Bemenet	1.	2.	3.	4.	5.	6.	Kimenet
Neuronok száma	3	15	30	60	120	240	480	5
Aktivációs függvény	tanh	tanh	tanh	ReLU	tanh	ReLU	tanh	tanh

4.2. Neurális hálózat alapú szerszámpálya generálás szimulációs vizsgálata

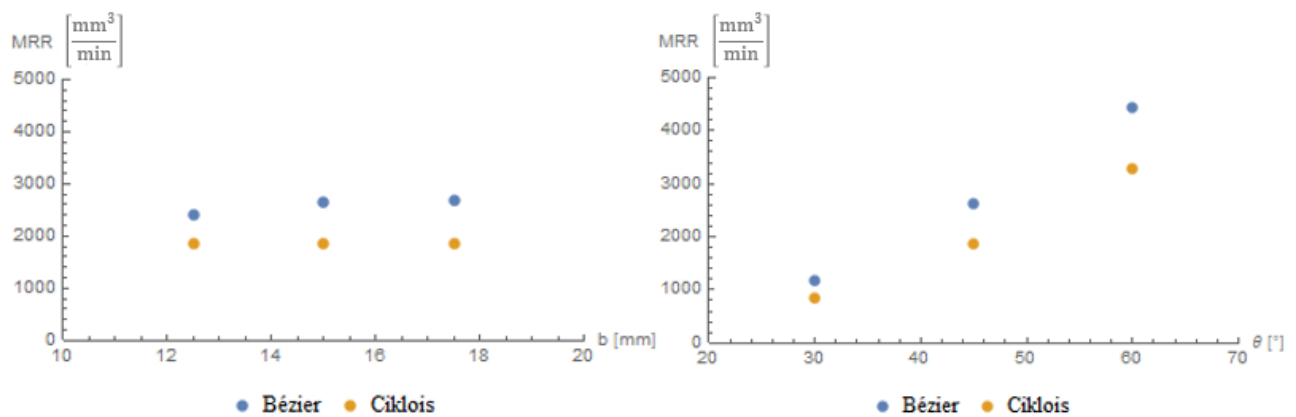
A betanított ANN jó közelítést ad a tanító adatkimenetekre, a validációs adatkimenetekre, valamint a korábban a prediktív modell által nem ismert olyan adatokra is, amelyek a tanító és validációs adatok által lefedett intervallumon helyezkednek el. Az ANN által adott kimenetek MRR értékei különböző horonyszélességekre, kontaktszögekre, és minimális görbületi sugarakra hasonlóak a kereső algoritmus által generált kimenetekre kapott értékekhez. Összehasonlítva az ANN, valamint a kereső

algorithmus futási idejét a forgácsolási kísérletek során alkalmazott bemeneti kombinációk esetén megállapítható, hogy az ANN futási ideje töredéke a kereső algoritmusénak (7. Táblázat).

7. Táblázat. Az ANN, valamint a kereső algoritmus futási idejének összehasonlítása.

Bemeneti kombináció			ANN futási idő [s]	Kereső algoritmus futási idő [s]
b [mm]	θ [°]	ρ_{min} [mm]		
12,5	45	0,15	0,0019943	234,88
15	45	0,15	0,0009666	340,13
17,5	45	0,15	0,0010021	110,84
15	30	0,15	0,0015073	50,93
15	60	0,15	0,0019995	339,77

A számított MRR értékek nagyobbak a korábban használt ciklois szerű szerszámpálya-szakaszokhoz viszonyítva. A ciklois szerű szerszámpályák MRR értéke nem függ jelentősen a horonyszélességtől, míg a Bézier görbe alapú szerszámpályák növekvő MRR értéket mutatnak a horonyszélesség növekedésével (10. a) Ábra). A kontaktszög növekedése mindkét típusú szerszámpályát befolyásolja, azonban a Bézier görbék esetében a növekedés jelentősebb (10. b) Ábra).



10. Ábra. a) Az MRR értékek változása a horonyszélesség (b) függvényében.

b) Az MRR értékek változása a kontaktszög (θ) függvényében

4.3. Forgácsolási kísérletek

Minden tesztelt szerszámpálya problémamentesen futott a szerszámgépen. A megmunkált hornyok a tervezetnek megfelelően elkészültek (11. Ábra).



11. Ábra. A kialakított hornyok.

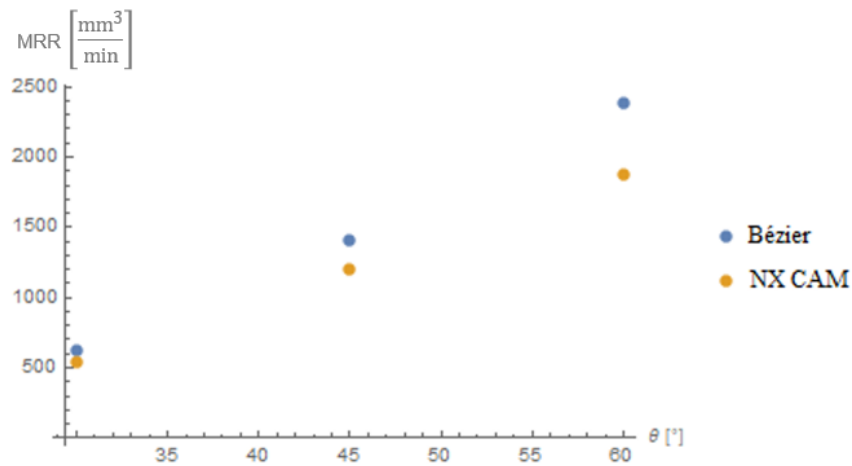
A szerszámgép vezérlése a forgácsolási időt 20 ms pontossággal mérte. A periódusidő – egy szerszámpálya periódus teljesítéséhez szükséges idő – segítségével került kiszámításra az MRR a forgácsoló és kapcsoló szakaszokra együttesen vonatkoztatva (8. Táblázat). A kísérletek során kapott MRR értékek hasonlóak az elméleti értékekhez.

8. Táblázat. A szerszámpályák tulajdonságai.

<i>A szerszámpályák tulajdonságai</i>							
Pályagenerálási módszer	b [mm]	θ [°]	ρ_{min} [mm]	Elméleti MRR $\left[\frac{mm^3}{min}\right]$	Mért MRR $\left[\frac{mm^3}{min}\right]$	Maximális erő [N]	
Kereső algoritmus	15	45	0,15	1419,184	1416,850	214	
ANN	15	45	0,15	1409,369	1406,269	204	
ANN	12,5	45	0,15	1248,127	1242,224	198	
ANN	17,5	45	0,15	1438,615	1436,483	209	
ANN	15	30	0,15	621,642	620,527	161	
ANN	15	60	0,15	2386,512	2381,655	272	
NX CAM	15	30	0,15	-	542,918	186	
NX CAM	15	45	0,15	-	1196,825	240	
NX CAM	15	60	0,15	-	1877,167	286	

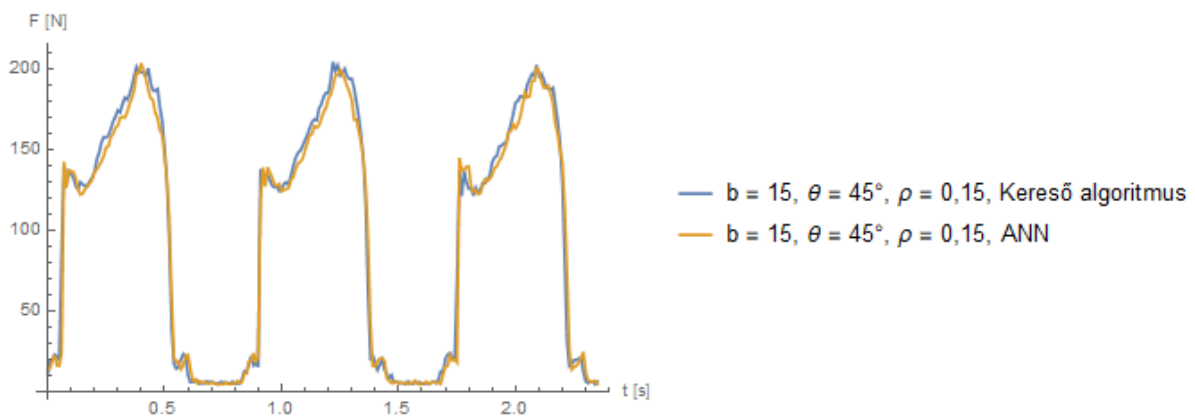
A szimulált és mért adatok közötti eltérés egyik esetben sem éri el a 0,5%-ot, az átlagos eltérés 0,23%. Ez alapján kijelenthető, hogy a mérések közel ideális körülmények között zajlottak.

Az MRR értékek alapján az azonos paraméterekkel rendelkező NX CAM szerszámpályák, illetve ANN szerszámpályák közül minden esetben az ANN szerszámpályák teljesítettek jobban. Az MRR különbségek a szerszámpálya családok között a kontaktszög növekedésével nőttek (12. Ábra).



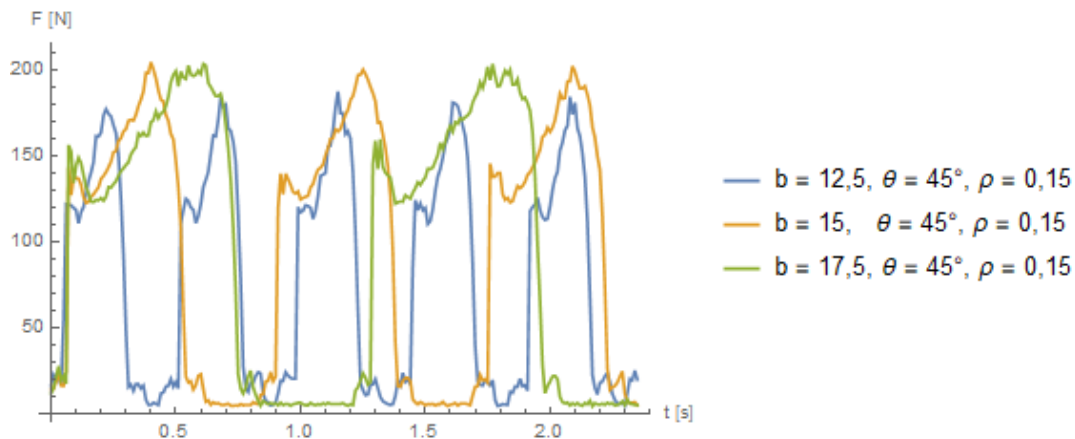
12. Ábra. A Bézier görbe alapú, valamint az NX CAM segítségével tervezett szerszámpályák MRR értékeinek összehasonlítása.

A forgácsolási erők az azonos paraméterekhez tartozó kereső algoritmus, valamint ANN által generált szerszámpályák esetében hasonlóak voltak (13. Ábra). A kereső algoritmus esetén nagyobb volt a maximálisan fellépő erő.



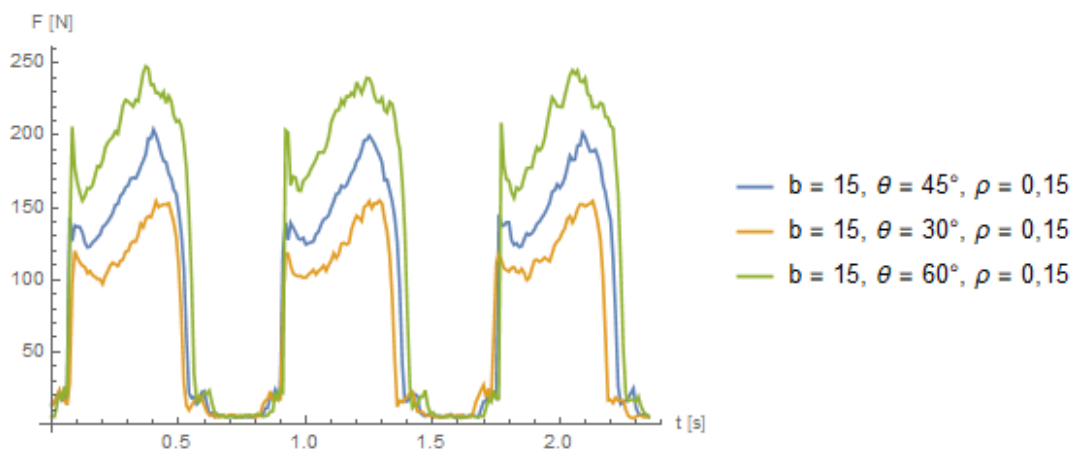
13. Ábra. Forgácsolási erőprofil a kereső algoritmus, valamint az ANN által tervezett szerszámpálya esetén. (b [mm]-ben, θ [°]-ban, ρ [mm]-ben értelmezendő.)

A Bézier görbe alapú szerszámpályák esetén megfigyelhető, hogy nagyobb horonyszélesség esetén az erők jellege elnyújtottabb, ami a hosszabb forgácsolási szakasszal, ebből következően a hosszabb forgácsolási idővel magyarázható. A horonyszélesség azonban nem befolyásolja jelentősen az erők nagyságát a kritikus pontokban (14.Ábra).



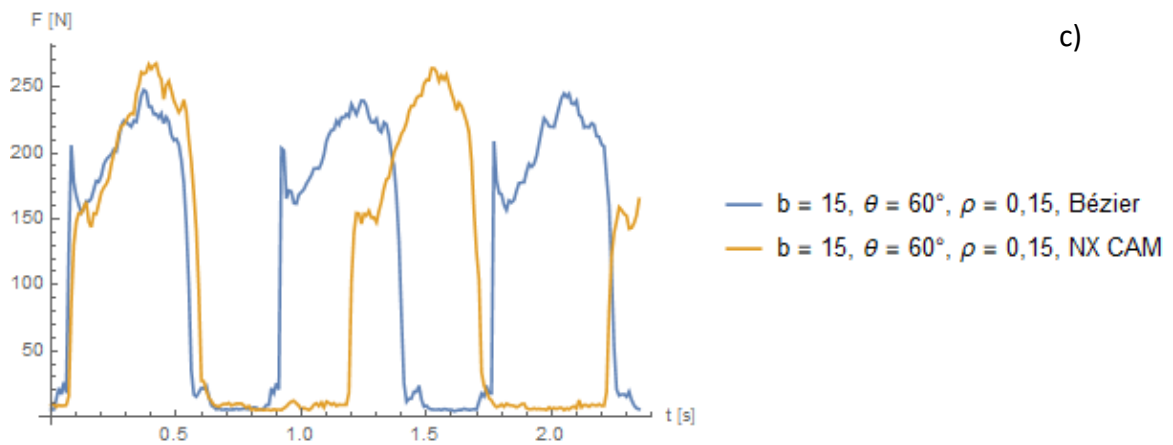
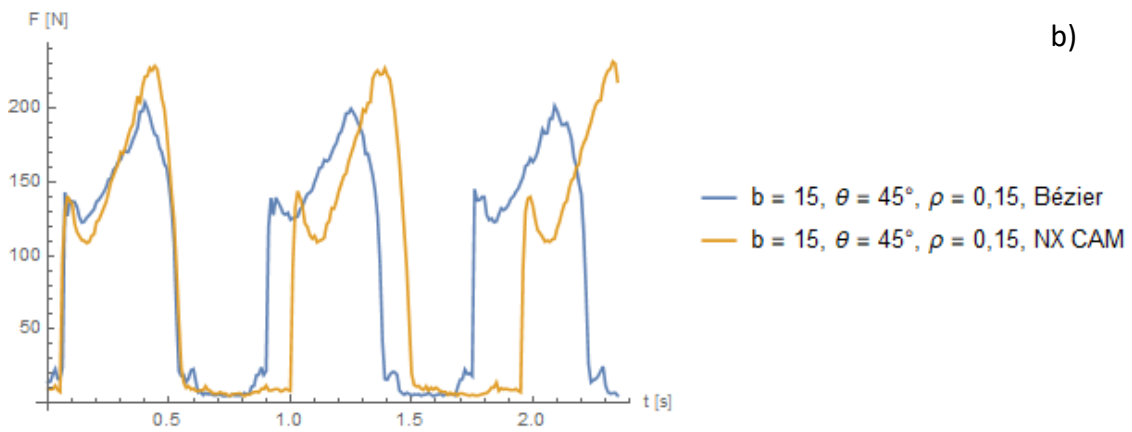
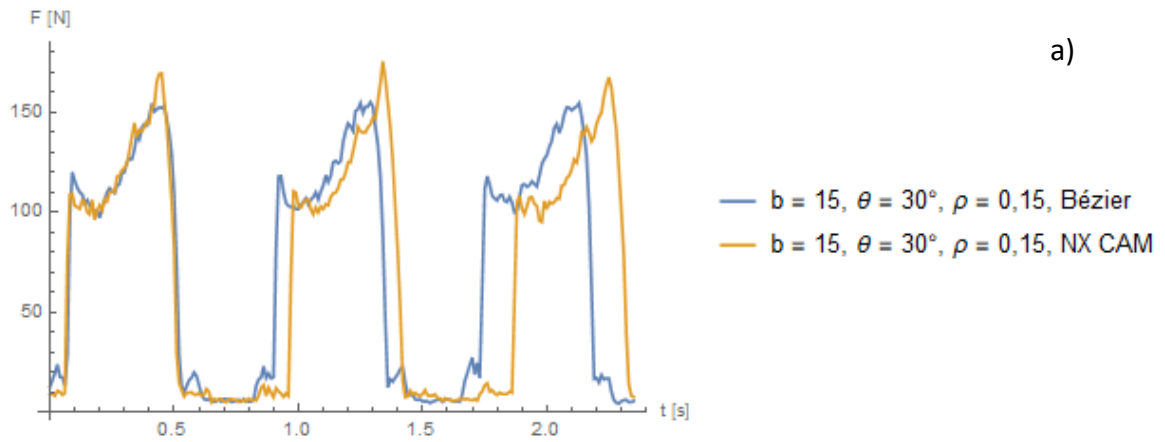
14. Ábra. Bézier görbe alapú szerszámpályák forgácsolási erőprofilja 45° -os kontaktszög esetén. (b [mm]-ben, θ [°]-ban, ρ [mm]-ben értelmezendő.)

A kontaktszög változásával a kritikus pontokban fellépő erők is változnak (15. Ábra).



15. Ábra. Bézier görbe alapú szerszámpályák forgácsolási erőprofilja 15 mm-es horonyszélesség esetén. (b [mm]-ben, θ [°]-ban, ρ [mm]-ben értelmezendő.)

A kísérlet során a Bézier görbe alapú szerszámpályák minden esetben jobbnak bizonyultak az NX CAM szerszámpályáknál a forgácsolási erőket tekintve (16. a), b), c) Ábra). Az erőkülönbség a vizsgált esetekben a kritikus pontokban átlagosan közelítőleg 30 N.



16. Ábra. Bézier görbe alapú, illetve NX CAM szerszámpályák forgácsolási erőprofilja
a) $\theta = 30^\circ$, b) $\theta = 45^\circ$, c) $\theta = 60^\circ$ esetén. (b [mm]-ben, θ [°]-ban, ρ [mm]-ben értelmezendő.)

4.4. Eredmények értékelése

A Bézier-görbe alapú szerszámpályák rugalmas megoldást nyújtanak trochoidális marás esetén. A vizsgált szerszámpálya szakaszok jobban kihasználják az előállítandó geometriából adódó lehetőségeket, mivel a pálya alakja a horonyszélességtől és a kontaktszögtől is jelentős mértékben függ, ellenben a korábban használt ciklois alapú pályákkal, amelyek esetében csak a kontaktszög változása befolyásolja jelentősen a fajlagosan leválasztott anyagmennyiséget.

A szerszámpálya generálási algoritmus kialakítása miatt a Bézier görbe alapú szerszámpályák forgácsoló és összekötő szakaszai is kielégítik a folytonosságra, valamint a minimális görbületi sugárra vonatkozó kritériumokat. A geometriai feltételek teljesülése miatt ezek a szerszámpályák tehát megfelelőek HSM alkalmazásokra.

Az ANN segítségével a szerszámpályák alapját képező Bézier görbék előállítása jelentősen gyorsabbá tehető a kereső algoritmussal összehasonlítva, mivel a nagy számítási kapacitást igénylő tanítási folyamat előre elvégezhető. A felhasználó által megadott aktuális bemeneti adatokhoz tartozó szerszámpálya azonnal generálható. Az ANN architektúra megválasztására nincs egzakt algoritmus, így empirikusan, különböző típusú és mélységű ANN-ek vizsgálatával került meghatározásra a feladat kritériumainak megfelelő prediktív modell. A statisztikai értékek azt mutatják, hogy az ötödfokú Bézier görbe kontrollpontok y koordinátáinak meghatározása komplex, nagy mélységű és neuron számú ANN-t igényel. Az egyszerűbb modellek csak bizonyos bemeneti adat intervallumokon képesek megfelelő pontosságú Bézier görbeszakaszokat definiálni. Ezek a bemeneti adat intervallumok az egyes ANN-ek esetében eltérőek.

A szimulációs eredményeket a kísérleti eredményekkel összehasonlítva kijelenthető, hogy a forgácsolás közel ideális körülmények között megy végbe, mivel nem tapasztalható jelentős különbség az elméleti és számított anyagleválasztási sebesség között. A kísérletek során mért erők kedvezőbbek voltak a Bézier görbe alapú szerszámpályák esetén. A fogásmélység növelésével az erők növekedése elméletben lineáris, azonban a helikális szerszámél-kialakítás jobb érvényesülése miatt a valós eset ennél kedvezőbb. A közel lineáris kapcsolatból viszont következik, hogy nagyobb fogásmélység esetén a tesztelt pályatípusok közti forgácsolási erőbeli különbségek is növekednek. A Bézier görbe alapú trochoidális marás szerszámkímélő tulajdonsága tehát nagyobb fogásmélység esetén jobban érvényesül.

5. ÖSSZEFOGLALÁS

A kutatás célja a gépi tanulás Bézier görbe alapú szerszámpálya tervezési célra történő használatának bevezetése volt. A kifejlesztett algoritmus validálása szimulációs és kísérleti úton történt, és más módszerekkel összehasonlítva jelentősen kedvezőbb eredményeket produkált.

A munka fő újdonságtartalma a Bézier görbék optimalizálására fejlesztett mesterséges neurális hálózat megalkotása. A számítások alátámasztották, hogy a készített program a korábban alkalmazott kereső algoritmustól nem marad el a pontosság terén. A futási idők összehasonlításából egyértelműen kiderült, hogy az új program nagyságrendekkel gyorsabb. Ennek köszönhetően a Bézier görbe alapú trochoidális marási stratégia alkalmassá válhat ipari alkalmazásokra.

A szimulációk azt mutatják, hogy a Bézier görbe alapú módszer ideálisabb megoldás trochoidális szerszámpályák definiálására, mint a korábban használt ciklois alapú, valamint az NX CAM által nyújtott megoldás. A forgácsolási kísérletek során a Bézier görbe alapú szerszámpályák esetében nagyobb fajlagosan leválasztott anyagmennyiség mellett kisebb forgácsolási erők voltak mérhetőek az NX CAM szerszámpályákhoz képest. Ez a szerszám élettartamának növelését is jelenti.

Összességében kijelenthető, hogy a Bézier görbe alapú szerszámpályák a bemutatott mesterséges neurális hálózat segítségével előállíthatók a gyártási környezetben elvárt gyorsasággal.

Kitekintés

Az NX CAM szerszámpályák előnye a bemutatott Bézier görbe alapú pályákhoz képest, hogy az előbbieket esetében a horony bekezdése lekerekített. Ez egyes esetekben megkönnyítheti a szerelést. A későbbiekben hasonló lekerekítési funkcióval, valamint simító ciklussal kiegészíthető a Bézier görbe alapú pályageneráló algoritmus is, mivel a fejlesztett módszer egy nagyoló ciklust eredményez. További fejlesztési lehetőség a számítási algoritmus kiterjesztése általános alakú horonygeometriák kezelésére, valamint az előtolás-szabályozás bevezetése a termelékenység további növelésére.

A megvalósítandó feladathoz illő aktivációs függvények segítségével az ANN komplexitása növelhető, azonban a számos lehetséges függvény és neurális hálózat típus minden kombinációjának tesztelése jelentős mennyiségű számítást igényel, így a kutatás során kísérleti alapon történt az aktivációs függvények megválasztása. Az aktivációs függvények további vizsgálatával csökkenthető a neurális hálózat rétegszáma, így a betanítás során igényelt memória és számítási igény, ami lehetővé tenné nagyobb tanító adathalmaz feldolgozását, kiterjesztve a szerszámpályatervező program felhasználási lehetőségeit.

6. SUMMARY

The aim of this study was to introduce a machine learning based approach to Bézier curve-based toolpath generation. The results were validated through simulations and milling experiments. The presented method showed superior results compared to the tested formerly used methods.

The main novelty of this study is the construction of the neural network developed to optimize Bézier curves. It was verified through computation that the constructed program is not less precise compared to the formerly used search algorithm. However, comparing the runtime of the neural network based program with the search algorithm, it can be concluded that the new program is faster by magnitudes. The fast computation makes the Bézier curve based trochoidal milling a suitable alternative for industrial applications.

The simulations verified that the Bézier curve-based approach is superior to the formerly used cycloid-based, and the NX CAM Adaptive Milling approach when it comes to defining trochoidal toolpaths. The milling experiments showed that the material removal rate was higher in the case of the Bézier curve-based toolpaths, while the cutting forces were smaller compared to the NX CAM toolpaths. This results in longer tool life.

In summary, it can be concluded that the Bézier curve-based toolpaths, generated using the presented artificial neural network, are able to provide the efficiency required by the industry.

The advantage of the NX CAM toolpaths compared to the Bézier-curve based ones is that in the case of the former, the sidewall of the slots are rounded. In some cases, this may facilitate assembly. The Bézier curve-based algorithm can be extended with a rounding feature, as well as with a finishing cycle, as the developed toolpath generation algorithm creates roughing cycles. Further possibilities are to extend the algorithm to general slot geometry, and to introduce feed-control to increase the productivity.

A well selected activation function increases the complexity of the ANN, however, testing each possible combination of the existing functions would result in significant computational cost, thus in the presented study, the selection of the activation functions was based on experimental data. Further studying of the activation functions may help in decreasing the number of layers of the neural network, resulting in less memory and computational cost during the training process. This would enable the processing of larger datasets, extending the capabilities of the tool path generation.

FELHASZNÁLT FORRÁSOK

- [1] Szalóki, I., Csuka, S., & Sipos, S. (2014). New test results in cycloid-forming trochoidal milling. *Acta Polytechnica Hungarica*, 11(2), 215-228.
- [2] Khatri, A., Jahan, M. P., & Ma, J. (2019). Assessment of tool wear and microstructural alteration of the cutting tools in conventional and sustainable slot milling of Ti-6Al-4V alloy. *The International Journal of Advanced Manufacturing Technology*, 105, 2799-2814.
- [3] Gnytko, O., & Kuznetsova, A. (2022). Theoretical research of the chip removal process in milling of the closed profile slots. *Archives of Materials Science and Engineering*, 113(2).
- [4] Gómez, G., De Lucio, P. F., Del Olmo, A., De Pissón, G. M., Jimeno, A., González, H., & De Lacalle, L. L. (2021, October). Comparison between milling roughing operations in full slotting manufacturing: trochoidal, plunge and conventional milling. In *IOP Conference Series: Materials Science and Engineering* (Vol. 1193, No. 1, p. 012003). IOP Publishing.
- [5] Avrampos, P., & Vosniakos, G. C. (2018). Coverage in planar surface polishing by trochoidal tool paths. *International Journal on Interactive Design and Manufacturing (IJIDeM)*, 12, 663-675.
- [6] Wagih, M., Maher, I., El-Hofy, H., Yan, J., & Hassan, M. A. (2023). Analysis and development of elliptical tool path in trochoidal milling. *CIRP Journal of Manufacturing Science and Technology*, 47, 168-183.
- [7] Patil, P., Polishetty, A., Goldberg, M., Littlefair, G., & Nomani, J. (2014). Slot machining of Ti6Al4V with trochoidal milling technique. *Journal of Machine Engineering*, 14.
- [8] Deng, Q., Chen, Z. C., Chang, Z., Shen, R., & Zhou, Y. (2020). A model for investigating the temperature of trochoidal machining. *Journal of Industrial and Production Engineering*, 37(4), 194-203.
- [9] Karkalos, N. E., Karmiris-Obratański, P., Kurpiel, S., Zagórski, K., & Markopoulos, A. P. (2021). Investigation on the surface quality obtained during trochoidal milling of 6082 aluminum alloy. *Machines*, 9(4), 75.
- [10] Abukhshim, N. A., Mativenga, P. T., & Sheikh, M. A. (2006). Heat generation and temperature prediction in metal cutting: A review and implications for high speed machining. *International Journal of Machine Tools and Manufacture*, 46(7-8), 782-800.

- [11] Osman, K. A., Yılmaz, V., Ünver, H. Ö., Şeker, U., & Kılıç, S. E. (2020). Slot milling of titanium alloy with hexagonal boron nitride and minimum quantity lubrication and multi-objective process optimization for energy efficiency. *Journal of Cleaner Production*, 258, 120739.
- [12] Gupta, K., & Davim, J. P. (Eds.). (2020). *High-Speed Machining*. Academic Press.
- [13] Thusty, J. (1993). High-speed machining. *CIRP annals*, 42(2), 733-738.
- [14] Schulz, H., & Moriwaki, T. (1992). High-speed machining. *CIRP annals*, 41(2), 637-643.
- [15] Zhuang, C., Xiong, Z., & Ding, H. (2010). High speed machining tool path generation for pockets using level sets. *International journal of production research*, 48(19), 5749-5766.
- [16] Jacso, A., Sikarwar, B. S., Phanden, R. K., Singh, R. K., Ramkumar, J., & Sahu, G. N. (2022). Optimisation of tool path shape in trochoidal milling using B-spline curves. *The International Journal of Advanced Manufacturing Technology*, 121(5), 3801-3816.
- [17] Held, M., & Spielberger, C. (2014). Improved spiral high-speed machining of multiply-connected pockets. *Computer-Aided Design and Applications*, 11(3), 346-357.
- [18] Oh, N. S., Woo, W. S., & Lee, C. M. (2018). A study on the machining characteristics and energy efficiency of Ti-6Al-4V in laser-assisted trochoidal milling. *International Journal of Precision Engineering and Manufacturing-Green Technology*, 5, 37-45.
- [19] Shixiong, W., Wei, M., Bin, L., & Chengyong, W. (2016). Trochoidal machining for the high-speed milling of pockets. *Journal of Materials Processing Technology*, 233, 29-43.
- [20] Otkur, M., & Lazoglu, I. (2007). Trochoidal milling. *International Journal of Machine Tools and Manufacture*, 47(9), 1324-1332.
- [21] Huang, X., Wu, S., Liang, L., Li, X., & Huang, N. (2020). Efficient trochoidal milling based on medial axis transformation and inscribed ellipse. *The International Journal of Advanced Manufacturing Technology*, 111, 1069-1076.
- [22] Gu, H., Wang, L., & Guo, K. (2024). A novel method for trochoidal milling tool path tailoring based on curvature variation. *Journal of Manufacturing Processes*, 119, 531-549.
- [23] Jacso, A., Lado, Z., Phanden, R. K., Sikarwar, B. S., & Singh, R. K. (2023). Bézier curve-based trochoidal tool path optimization using stochastic hill climbing algorithm. *Materials Today: Proceedings*, 78, 633-639.

- [24] Xu, K., Wu, B., Li, Z., & Tang, K. (2019). Time-efficient trochoidal tool path generation for milling arbitrary curved slots. *Journal of manufacturing science and engineering*, 141(3), 031008.
- [25] Eluyode, O. S., & Akomolafe, D. T. (2013). Comparative study of biological and artificial neural networks. *European Journal of Applied Engineering and Scientific Research*, 2(1), 36-46.
- [26] Shanmuganathan, Subana. (2016). *Artificial neural network modelling: An introduction*. Springer International Publishing.
- [27] Sharma, Sagar, Simone Sharma, and Anidhya Athaiya. "Activation functions in neural networks." *Towards Data Sci* 6.12 (2017): 310-316.
- [28] Diederik P. Kingma, Jimmy Ba. (2015). Adam: A Method for Stochastic Optimization. Conference for Learning Representations.
- [29] Deisenroth, M. P., Faisal, A. A., & Ong, C. S. (2020). *Mathematics for machine learning*. Cambridge University Press.
- [30] Hartmann, C., Opritescu, D., & Volk, W. (2019). An artificial neural network approach for tool path generation in incremental sheet metal free-forming. *Journal of Intelligent Manufacturing*, 30(2), 757-770.
- [31] Balic, J., & Korosec, M. (2002). Intelligent tool path generation for milling of free surfaces using neural networks. *International Journal of Machine Tools and Manufacture*, 42(10), 1171-1179.
- [32] Jacso, A., Szalay, T., Sikarwar, B. S., Phanden, R. K., Singh, R. K., & Ramkumar, J. (2023). Investigation of conventional and ANN-based feed rate scheduling methods in trochoidal milling with cutting force and acceleration constraints. *The International Journal of Advanced Manufacturing Technology*, 127(1), 487-506.
- [33] National Instruments (2024). The Ni TMDS file format. <https://www.ni.com/en/support/documentation/supplemental/06/the-ni-tdms-file-format.html?srsltid=AfmBOooo2tT-15GvGfnpX7oMLswIA31S8r0iTWChFs--eDa0lrTOgPem> (2024.08.23.).
- [34] National Instruments (2022). Documentation, TMSD file format. <https://www.ni.com/en/support/documentation/supplemental/07/tdms-file-format-internal-structure.html?srsltid=AfmBOoobV2b8Sc4HUXhqx5fybUTlJxOzKkiVYLw9tneMRXk9OehkInoH> (2024.08.23.).