



Budapesti Műszaki és Gazdaságtudományi Egyetem
Gépészmérnöki Kar
Gyártástudomány és -technológia Tanszék

**5 tengelyes megmunkálás ipari robottal: a
programozás szoftveres támogatásának lehetőségei
és laboratóriumi kísérletek**

TDK dolgozat

Készítették:

Nyemcsok Máté

ZC1GHV

BS.c. hallgató

Rostási Gergő

JK0NB1

BS.c hallgató

Konzulens:

Tóth András

Tudományos Munkatárs

Budapest, 2014

KÖSZÖNETNYILVÁNÍTÁS

Ezúton szeretnék köszönetet mondani Tóth András konzulensünknek a dolgozat során adott hasznos ötletekért és jó tanácsokért. Köszönettel tartozunk a Delcam Hungary Kft-nek, hogy liszensz jogot biztosítottak a PowerMill CAM szoftver használatához a dolgozatunk során, illetve Tóth Péternek a szoftverben nyújtott segítségéért. Köszönjük a Gyártástudomány és –technológia Tanszéknek, hogy hozzáférést biztosítottak az Siemens NX, és Tecnomatix Process Simulate szoftverekhez. a GraphIT Kft –nek és Sántha Csabának a szoftverhasználattal kapcsolatos tanácsokért.

Köszönjük Kocsis Imre tanszéki főmérnöknek, hogy segítséget nyújtott a koordináta mérőgépes bemérés során, illetve Juhász Imre, és Szűcs László tanszéki dolgozónak az adapter tárcsák elkészítéséért.

Továbbá köszönjük Jurák Mihálynak a szoftveres rész elkészítésében nyújtott tanácsokért, illetve Fehér Zoltán MSc hallgatónak a robot beüzemelésében nyújtott segítségért.

Tartalom

Köszönetnyilvánítás	2
1. Bevezető	5
2. Szakirodalmi áttekintés	6
3. Munkakörnyezet bemutatása	25
3.1. Kísérleti elrendezés	25
3.1.1. ABB IRB 140	25
3.1.2. ABB IRB 1400h	26
3.1.3. Főorsó, és szerszám	26
3.1.4. JR3 erőmérő szenzor	26
3.2. Virtuális munkakörnyezet, szoftverek	27
3.2.1. Siemens NX 8.5	27
3.2.2. Siemens Tecnomatix RobotExpert / Process Simulate	27
3.2.3. Delcam PowerMill Robot Interface	28
4. Robotok programozása marási műveletekre	29
4.1. Siemens NX 8.5 & Tecnomatix Process Simulate esetében	29
4.2. Delcam PowerMill	32
4.3. Rapid Program elkészítése	34
4.4. Összesítés	35
5. Laboratóriumi kísérletek	37
6. Robotos marás alkalmazási lehetőségek	47
7. Fejlesztési lehetőségek	49
8. Összefoglaló	50
9. Források	51
10. Melléklet	54

Jelölések jegyzéke

Rövidítések	
Rövidítés	Megnevezés
PS	Siemens Tecnomatix Process Simulate
OLP	Offline Programming
CLS	Cutter Location Source (File)
TCP	Tool Center Point – Robot vezérelt pontja
via	A robotprogram köztes pontjai
mtd	Machine Tool Data
CAD	Computer-aided design
CAM	Computer-aided manufacturing
HDCM	High Dynamics Compensation Mechanism
COMET	Components and Methods
PUR	Poliuretán hab
EPS	Expandált polisztirol

1. Bevezető

Napjainkban az anyagleválasztással járó gyártási folyamatokban meghatározó szerepe a CNC vezérlésű szerszám gépnek van [1]. Fő ismervük a pontosság és a megbízhatóság. Viszont legnagyobb hátrányuk a méretükhöz viszonyított kis munkatér. Ebből adódóan egyes alkatrészeket csak több műveletben képesek készre munkálni, ezáltal lassul a gyártási folyamat. Továbbá az új termék gyártása előtt a gépbeállítási műveletek is időigényesek, ami rontja a rugalmasságukat. A jelenlegi ipar rugalmas gyártóeszközeit az ipari robotok jelentik [2].

Egy ipari robot többféle munkavégzésre is képes, mint például: anyagmozgatás, hegesztés, festés, mérés, vágás stb. További nagy előnyük, hogy a méretükhöz képest nagy munkatérrel rendelkeznek, valamint a gépi költségük 2-5-szer kevesebb a szerszám gépekhez képest [3]. Felismerve a robotokban rejlő lehetőségeket, megmunkálási feladatokban is elkezdték alkalmazásukat. A legnagyobb előnyt itt is a relatív nagy munkatérük jelentette. Alkalmazásukkal gyakorlatilag egy befogásból is hozzáférhető egy alkatrész befogó készülék által szabadon hagyott minden oldala valamint 5 tengelyes megmunkálás is egyszerűen elvégezhető.

Az ipari robotok megmunkálási célra való használata nem új keletű dolog. Már a 80-as évek végén is kerültek feljegyzésre robotos polírozással és sorjázással kapcsolatos tudományos vizsgálatok. Ezeken a területeken az ipari robotok alkalmazása ma már egy elterjedt technológiának mondható [1].

Marási műveleteknél is rendkívül jól kihasználhatóak az ipari robot egyedi tulajdonságai. Ebben az esetben viszont a bonyolult szerszámpályák programozása, a robot szerkezetének gyenge merevségi tulajdonságai, az ipari robotokra jellemző csuklós kialakítás okozta konfigurációs és szingularitási problémák, munkatérbeli tulajdonság anizotrópia, irányváltási problémák olyan kihívások elé állították a mérnököket, melyeknek a megoldását csak az elmúlt néhány évben sikerült megtalálni.

Dolgozatunkban mi is az ipari robotok marási feladatokban történő alkalmazásával, és annak szoftveres támogatásával foglalkozunk.

2. Szakirodalmi áttekintés

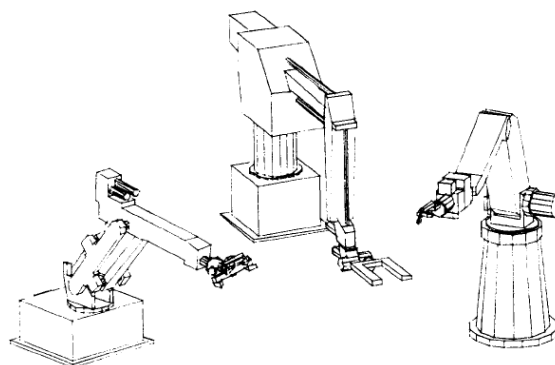
A szakirodalmi áttekintésben átfogóan ismertetjük az ipari robot alkalmazhatóságát, és annak korlátait marási feladatokban. Ismertetjük a már eddig elért kutatások, kísérletek eredményeit, melyek a megmunkálási pontosság javítását célozták meg. Bemutatjuk a szerszámpályák programozásának szoftveres lehetőségeit.

2.1. A technológia kifejlődése

Ebben a részben azokat a kutatásokat ismertetjük, amelyek nagyban hozzájárultak a technológia fejlődéséhez.

2.1.1. Szimuláció és programozás

Az első írásos feljegyzések robotos alkalmazások virtuális környezetben történő megjelenítéséről az 1980-as évek elejétől kezdve jelentek meg. 1983-ban Mc Bonney és társai az Egyesült Királyságban kifejlesztették a GRASP (Graphical Robot Applications Simulation Package) nevezetű szoftvert. A GRASP, mint ahogyan majd az őt követő szoftverek nagy többsége is, a Denavit-Hartenberg-féle transzformációval építette fel a robotok kinematikai modelljét. A programon belül lehetett a robot és a munkatér modelljét is elkészíteni. A modellek felépítése primitív formákból történt (1. ábra). A már egyszer letesztelt robotmodellek, egy könyvtárba kerültek, ami lehetővé tette további cellákba történő beépítését is [4].



1.ábra: GRASP segítségével, primitívekből előállított robotmodellek.[4]

Nem csak a robotok, de egy komplett cella is elmenthető volt, amelynek módosításával egyszerűen lehetett új cellaelrendezést létrehozni. A virtuális robot mozgatása többféleképpen is kivitelezhető volt. Az egyszerűbb eset, amikor a robotcsuklók egyenkénti mozgatásával

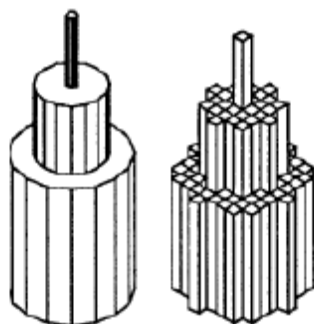
változtatták a robot helyzetét. Ennél egy kényelmesebb megoldás, hogy megadható volt egy pont, és kiszámolva a csuklósögeket a robot felvette a pozíciót. A pont meghatározható volt beírt koordinátákkal, grafikusan vagy valamelyik elemet referenciaként használva. Sorrendben eltárolt pontok szerszámpályát alkottak. A pálya menti robotmozgás szimulálható volt, ezáltal ki tudták szűrni az esetleges ütközéseket. A szimuláció után a szoftver képes volt a mozgáspálya posztprocesszálásra is. Előre beállítható volt a szerszám vezérelt pontjának sebessége, sőt egyéb funkciók utasításait is be lehetett illeszteni, mint pl.: hegesztés be/ki. Az elkészített programokkal számos sikeres gyakorlati kísérletet hajtottak végre. Azt meg kell jegyezni, hogy mivel a virtuális modell nem egyezett teljesen a valósággal, az elkészített programokon mindig kellett módosítani [4].

Az első, robotokkal megvalósított marási kísérletek 90-es évek elejétől kezdődően kerültek feljegyzésre. Robotos marási feladatok off-line programozásáról viszont viszonylag kevés írás jelent meg eddig. Ellenben a CAD modell alapján történő CNC pályatervezésről több ezer cikk és kutatás lett prezentálva az elmúlt 30 évben. Talán ez is lehet az oka, hogy a robotos megmunkálás programozásával kapcsolatosan kevesebb kísérlet indult, mivel sokan azt gondolják, hogy a kettő egyezik egymással. Ez viszont nincs teljesen így. Általában marási feladatokhoz 6 rotációs tengellyel rendelkező ún. humanoid robotot alkalmaznak. Ebben az esetben a robot munkaterének vannak olyan pontjai, melyek különböző konfigurációkban (különböző kareltrendezéssel) érhetők el. Viszont egy megmunkálási folyamaton belüli konfigurációváltás igen káros következményekkel járhat, melyek veszélyeztetik a robot és annak környezete épségét is. Lehetnek olyan pontok is egy szerszámpályán, ahol nincs egyértelmű megoldása az inverz kinematikának, ami végtelen sok megoldást eredményez. Ezt nevezik szingularitásnak. Továbbá, a robotok munkaterét gyakran külső tengelyek hozzáadásával bővítik. Az előzőleg felsorolt 3 példából kifolyólag egy hatékony off-line robotprogramozó szoftvernek kezelnie kell a konfigurációk beállítását, a szingularitást, valamint a külső tengelyek vezérlését is. Ezeket a feladatokat egy „hagyományos” CAD/CAM szoftver nem tudja ellátni [5].

Az elsők között 1990-ben Suh és Lee végzett robotos marási kísérletet. Ők elsősorban NC pályagenerálással foglalkoztak, de néhány fejlesztésüket roboton is tesztelték. Egyik munkájuk során szabadfelületekkel határolt zsebek marását végezték ipari robottal, nem-fémes anyagokon. A konvex és konkáv szabadfelületeket vonalakkal, körívekkel és görbékkel határoltnak tekintették. A megmunkálást nagyoló és simító ciklusra bontották. Ekkor még nem állt rendelkezésre olyan szoftver, mely segíthette volna a robotok programozását. Nagy problémát jelentett a megmunkálás közben előforduló ütközések kiszűrése [6].

1994-ben Huang és Oliver egy olyan módszert mutattak be, mely képes volt görbék interpolációjára, alámarás ellenőrzésére valamint megmunkálás közben a szerszámpálya paraméterek változtatására [7].

1995-ben Chen és Hu nagyoló megmunkálásokhoz alkalmazható programozási módszert dolgozott ki. A módszer lényege, hogy a modellt meghatározott rácsméretű hálózta be, ami meghatározta a megmunkált darab kontúrját (2. ábra). A rácsmérettel befolyásolni lehetett a megmunkált felület minőségét, viszont a túl finomra állítása, annyira megnövelhette a program nagyságát, hogy kezelése nehézséget jelentett néhány robotvezérlés számára [8].



2.ábra: Az eredeti és a leképzett alkatrész [8]

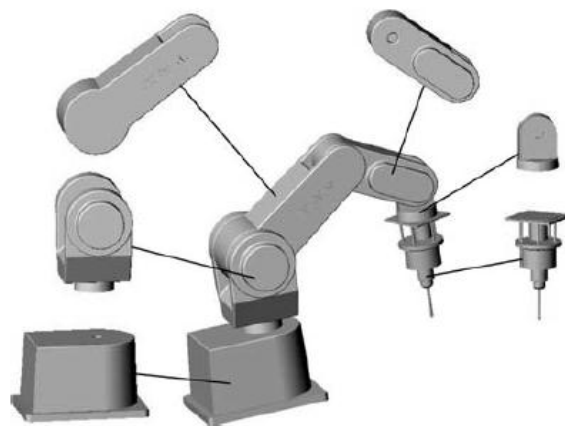
Az első publikált robotos marócella ipari célra való alkalmazása Vergeest és Tangelder nevéhez fűződik (1997). Ők prototípus készítésre próbálták a technológiát használni, bár cikkükben még nem esik off-line robotprogramozásról szó [9].

1999-re Song és Chen egy CAD alapú programot hoztak létre, mely képes volt a forgácsoló szerszámpálya generálása közben előre kiszűrni a lehetséges ütközéseket, majd a szerszámtengely döntésével elkerülni azokat. A pályageneráláson túl a robotprogram előállítását is elvégezte. A csuklók elfordulását a Denavit-Hartenberg szabályok szerint számolta ki. Az ütközésmentes szerszámpálya elkészítéséhez a programban 0,1mm-rel rövidebb szerszám lett definiálva, mint a valós esetben. Generálás közben program folyamatosan számolta a szerszám és munkadarab áthatását a megmunkálási pontokon kívül. Ha átfedést jelzett, a szerszámtengely egy kúp mentén történő mozgatása révén egy olyan pozícióba lett forgatva, ahol nincs ütközés. A kúp csúcsa a szerszám és a munkadarab érintkezési pontja volt [10].

Az ipari robotok CAD rendszer alapú off-line programozási lehetősége és a feladatok virtuális szimulációja óriási előre lépést hozott a termelékenység tekintetében. Segítségükkel már a tervezési szinten kiszűrhetők a robot hozzáférhetőségi, ütközési és egyéb nehézségei. A

grafikus felületen történő megjelenítés különböző modelltípusokon is végrehajtható, úgymint: drótváz modell, test modell. A modellek és a hozzárendelt algoritmusok segítségével ütközés vizsgálat, a robot kinematikai és dinamikai viselkedésének vizsgálata végezhető el.

2004-ben Mitsi és társaitól jelent meg egy cikk, amelyben az ipari robotok off-line programozásával kapcsolatos munkájukat ismertetik. Kutatásuk során egy öttengelyes (RV-M1) és egy hattengelyes (RV6) ipari robotot programoztak off-line módon. A robot, szerszám, munkadarab és a megmunkálási környezet SolidWorks-ben került modellezésre és megjelenítésre (3 *ábra*) [11].



3. ábra: SolidWorks-ben modellezett 6-tengelyes ipari robot [11]

Mitsi-ék egy SolidWorks-szel 'kooperatívan' működő külső szoftvert fejlesztettek ki. Röviden összefoglalva: az általuk kifejlesztett programban definiálhatóak a pályapontok, melyek alapján Denavit-Hartenberg transzformációs módszerrel meghatározzák a robotcsuklók szögeit. A szögek ismeretében a mozgást SolidWorks-ben jelenítették meg és végezték el az ellenőrzést. Ha a szerszám-pálya megfelelő a szoftver elkészítette az NC-programot [11].

Kicsit bővebben a szoftverről: A modellezés során definiált geometriai és kinematikai adatok, mint például: méretek, csukló határszögek, Denavit-Hartenberg paraméterek egy adatbázisba kerültek be. Ezek alapján történtek az inverz kinematikai számítások. Mozgástervezést durva és finom pályatervezésre osztották[11].

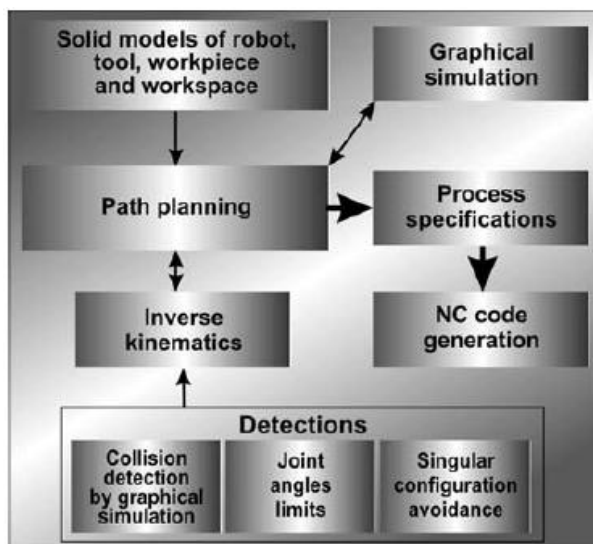
A durva tervezés során került meghatározásra a művelet megközelítési és visszahúzási mozgása is. Az ütközésmentes pályagenerálás „feltételezés és tesztelés” elven működött. A felhasználó megadhatta a pontok számát a kijelölt szerszám-pályán, valamint az interpoláció módját is (lineáris vagy spline) a köztes pontok számításához. A program minden pontban kiszámolta csuklószögeket, figyelembe véve a végállásokat, és ütközést keresett. Ha talált,

egy új pálya került kijelölésre. A folyamat addig tartott, amíg nem lett ütközésmentes a szerszámpálya [11].

A finom pályatervezés állítja elő a megmunkálás pályáját. A művelet alatt a robotnak a szerszámot megfelelő irányban, távolságban kell tartania és állandó sebességgel kell mozgatnia. Mivel egyes orientációk ütközéshez vezethetnek a felhasználó szabadon módosíthat, hogy a mozgás akadálymentes legyen. A finom pályatervezésnél lineáris interpolációval történik a köztespontok számítása [11].

Az elkészített szerszámpálya elmentésre kerül, és a gyakorlati beüzemelés előtt virtuálisan ellenőrizhető [11].

Az NC program generálásakor a robotmozgás programsorain kívül, számos egyéb utasítás is a programba illeszthető volt: pl.: hűtő-kenő folyadék ki-be kapcsolás, szerszámcsere megmunkálásnál vagy akár hegesztésnél az ív feszültség és -áramerősség értékek, huzal előtolás sebesség stb.. Az elkészített programot át lehetett tölteni a robotvezérlőbe. Az NC program generálás folyamatábrája a 4. ábrán látható. A pályatervezés Visual Basic-ben történt. A szimuláció az inverz kinematikai számításokkal párhuzamosan ment végbe. A kinematikai számítások Fortran programozási nyelven lettek megírva. A művelet összes paraméterét összegyűjtve elkészült az NC program [11].



4.ábra: NC program generálásának folyamata [11]

Szoftver szimulációs részének fejlesztésénél a kezelhetőség volt a fő szempont. Minden, a szoftver kezelőfelületén felvett pont vagy módosítás, a SolidWorks grafikus felületén is megjelent. A szimuláció pontossága érdekében, az interpolált mozgások is szemléltetve lettek. A mozgáspálya definiálásra négy lehetőség is volt. Megadható volt előre megszerkesztett

pályával, pályapontok egyenkénti beírásával, a munkadarab kulcspontjainak és eleinek kiválasztásával illetve szerkeszthető volt SolidWorks-ben vonalakból és spline-okból is. A pályapontok egyenként ki lehetett választani, a koordináták utólagosan is módosíthatóak voltak valamint a szerszám dőlése is állítható volt egy adott pontban [11].

Mitsiék módszere inkább a kevesebb pontot tartalmazó szerszámpályák szimulációjára és off-line programozására volt használható.

2008-ban a Robotmaster nevezetű off-line robotprogramozó szoftver, CAD/CAM alapú pályagenerálást tett lehetővé robotok részére is. Maga a program a Mastercam CAM szoftver integrált része. A pályagenerálást a CAM szoftver végezte, a robot specifikus beállításokat, szimulációt, posztprocesszálást pedig a Robotmasterben lehetett végrehajtani [12].

2.1.2. Ipari robot pontossága és annak javítását célzó kísérletek

A robotok pontossága nagyban függ az őket felépítő alkatrészek tűréseitől, illesztésétől, a használat okozta kopástól és a szerkezet statikus és dinamikus merevségétől. Növelhető a pontosság, ha az előzőleg felsorolt hibaforrások miatti rendszeres hiba mértéke meghatározható, ezáltal a robotpozicionálás is korrigálható. A kalibrálási folyamatok két fő típusra oszthatóak: modell alapú paraméteres- és modell alapú nem paraméteres kalibrációra [13].

A legtöbb modell alapú paraméteres kalibráció lényegében kinematikai modell-alapú kalibráció. Ennél a módszernél geometriai tényezők alkalmazásával épül fel a modell. Négy fő lépésből áll: modellezés, mérés, érvényesítés, korrekció [13].

Modellezés: A kinematikai modell a robot geometriájának és mozgásának matematikai leírása. A legismertebb a Denavit-Hartenberg féle homogén transzformációs módszer. Lényege, hogy minden egyes csuklóhoz hozzárendel egy koordináta-rendszert, és sorra leírja az egyes koordináta-rendszerek közötti transzformációs mátrixot. Ezzel az egész kar pozíciója megadható. A módszer hátránya, hogy nem szűrhetőek ki a szinguláris pontok. Ezért többen is kísérleteztek a D-H transzformáció módosításán, vagy egy teljesen új módszer kidolgozásán. Hayati például a D-H transzformáció módosítását ajánlotta, miszerint a párhuzamos tengelyek leírásánál egy további rotációs paraméter is kerüljön be a mátrixba. Teljesen eltérő megoldás volt például az S-modell, mely hat paraméterrel írt le minden egyes csuklót. Mivel a D-H féle leírás jellemző a legtöbb kinematikai modellnél, szinte az összes létező vezérlés is ezzel számol [13].

Mérés: Ez a legidőigényesebb folyamata a kalibrációnak, de mérés és hozzáilleszkedő rendszer nélkül egy jó modell használhatatlan. Lényege, hogy a szerszámközepont aktuális

mért helyzetét összehasonlítva az elméleti modell alapján számolt aktuális helyzettel megkapható a helyzethiba. Többféle módszer és műszer is használható, a bemérés elvégzésére: akusztikus érzékelő, kamerás rendszer, koordináta mérőgép, vizuális vagy automatikus teodolit [13].

Érvényesítés: Az érvényesítés folyamata numerikus módszerekből áll, melyekkel a szükséges paraméterek pontosan meghatározhatóak. A mérési eredmények alapján lineáris vagy nemlineáris legkisebb négyzetek módszerével közelítik meg a kinematikai hiba értékét [13].

Korrigálás: Az utolsó és egyben a legmeghatározóbb lépése a kalibrálási folyamatnak. Itt történik meg az új modell létrehozása a robot pozicionálását végző szoftverben. Mivel a kalibrált robot inverz kinematikája nem számolható ki analitikusan, numerikus algoritmust alkalmaznak, mint például Newton-Raphson algoritmus. Általában azt a módszert alkalmazzák, hogy a Descartes koordinátákkal megadott hibaértékekből, kiszámolja a csuklók korrekciós értékét. Az algoritmus hátránya, hogy on-line kompenzálás esetén nagy a számítási igény valamint szinguláris pont közelében az algoritmus leáll [13].

A kinematikai modell alapú kalibrációnál a robotcsuklók tökéletesen merevnek tekintettek. Ha robotot működése közben dinamikus igénybevétel is terheli, akkor az előzőleg ismertetett módszerrel felépített modell viselkedése jelentősen eltérhet a valóságtól. Ilyenkor a pontosságot nagyban befolyásolja a csuklók és tagok rugalmassága, teljesítmény átviteli elemek holtjátéka és hatásfoka, hőhatás. Ezek figyelembe vételéhez nem-kinematikai modellen alapuló kalibrációt kell alkalmazni. A nemlineáris modell megvalósítását tovább bonyolítja, hogy a mozgatott tömeg szög- és lineáris sebességének, - gyorsulásának dinamikai hatását is számításba kell venni. Jean és társai meghatározták, hogy a robot pontatlanságának a csuklók és a karelemek körülbelül 8-10%-át, a holtjáték 1-2%-át, míg a hő okozta deformáció 0,1 %-át okozza. A megmaradt körülbelül 88% a szerkezeti elemek gyártási pontatlanságából fakad [13].

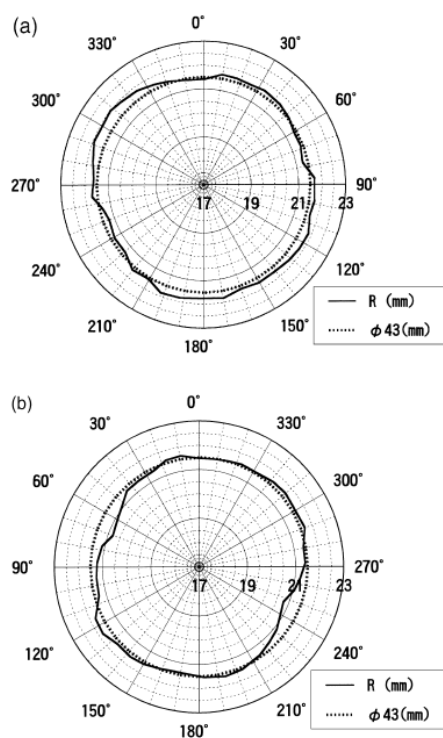
Jelenleg nincs elterjedt módszer a nem-kinematikai alapú modell felépítésre. Viszont az nyilvánvaló, hogy megmunkáló feladatoknál erős, dinamikus igénybevétel is terheli a robotot. Ezért a jelen kutatásainak fő célja az ipari robot dinamikus hatásokkal szembeni ellenálló képességének növelése. A későbbiekben ismertetünk néhány fejlesztést, amelyek gyakorlatban is igazolták hatékonyságukat [13].

A már korábban említett Vergeest és Tangelder féle marócellával legyártott prototípusok megmunkálás utáni pontossága 0,5-2 mm között mozgott [9].

1993-ban Morris és társai a robot kalibrációját koordináta mérőgéppel végezte el. A kísérlet során néhány pozíció pontos helyzetét koordináta mérőgéppel határozták meg és a mért értékek alapján felépítették a robotkar kinematikai modelljét. A világ- és szerszám koordináta-rendszer közötti kapcsolatot is létrehozták [14].

A robotkar deformációját leginkább a forgácsolási erők befolyásolják. Ennek érdekében a legtöbb kutatásban olyan anyagokkal kísérleteztek, melyek megmunkálásakor nem ébrednek nagy forgácsolási erők.

1996-ban Oki és Kanitani alumíniumon végzett nagysebességű megmunkálást. Kutatásuk során megállapították, hogy nagysebességű megmunkálásnál a fordulatszám frekvenciája és a robot sajátfrekvenciája olyan távol van egymástól, hogy nem lép fel rezonancia veszélye. Megfigyelték, hogy a forgácsolás iránya is hatással van a robotkar deformációjára. Kör alakú munkadarabot ellenirányú- és egyenirányú marással munkáltak meg. Ellenirányú marásnál a munkadarab átmérője nagyobb, míg egyenirányú marásnál kisebb lett az előírt méretnél (5. ábra) [15].



5.ábra: Megmunkált köralak keresztmetszete.
a) ellenirányú marás; b) egyenirányú marás [15]

A 1999-ben Matsuoka és társai tovább vizsgálták Oki és Kanitani kísérletét. Munkájuk során 3mm átmérőjű síkmaróval, nagysebességű megmunkálást hajtottak végre. A főorsó fordulatszáma 100000 1/min volt, munkadarabnak pedig extrudált, üreges, alumínium

ötvezeteket használtak. Az eredmények viszonyításához a megmunkálást hornyoló marón is elvégezték ugyanazokkal a vágási paraméterekkel. Méréseik szerint a robot esetében 50-70% kisebb erők terhelik főorsót, mint a hornyoló marónál. Ennek oka, hogy a robot szerkezetének deformációja elnyeli az ébredő forgácsolási erőket egy részét. Ha a főorsóra kisebb erők hatnak, nagyobb előtolás valósítható meg, ezáltal a robot forgácsolási teljesítménye is 60-70%-kal nagyobb volt [16].

A dinamikai kísérletek itt is alátámasztották, hogy a robot alacsony sajátfrekvenciája miatt a nagysebességű megmunkálásnál (100000 1/min-nél a frekvencia 1670 Hz) nem alakul ki rezonancia. A megmunkálás rezgések és erőingadozások ellen stabilnak mondható volt. A hornyoló maró első sajátfrekvenciája 1350 Hz, ami kb. 80000 1/min fordulatszámnál alakul ki, ezáltal fent áll a rezonancia kockázata. A statikus vizsgálatok kimutatták, hogy a robot merevsége 700-szor kisebb volt a mint a szerszámgépé (6.ábra). Látható, hogy az x irányú merevség 4-szer kisebb az y irányban mérttel. Ez magyarázza az Oki és Kanitani kísérleténél szemléltetett kör keresztmetszet elliptikus alakját. Ez alapján elmondható, hogy az ipari robot pontosságának javításánál létfontosságú szempont az x irányú erők csökkentése [16]

Table 3
Static stiffness of each of the machine tools

Machine tool	X ($\mu\text{m/N}$)	Y ($\mu\text{m/N}$)	Z ($\mu\text{m/N}$)
Fluting machine	0.13	0.08	0.09
Robot (PA-10)	83.65	20.35	68.76

6.ábra: A Fluting machine és a robot merevségi értékei x, y és z irányban[16]

2000-ben Young és Pickin lézeres interferométer segítségével mérte meg három különböző gyártó korszerű, 6-tengelyes ipari robotjainak pontosságát és ismétlési pontosságát. 5 pontot vettek fel a szerszám középponttal az x- és az y tengely mentén is, és mérték a helyzetpontosság hibáját a tengelyek irányában és arra merőlegesen is. Az eredmények szerint a tengelyre merőleges eltérés 0,2 mm-en belül volt mind a két irányban, míg a tengely mentén a hiba elérhette az 1,6 mm-t is [17].

1997-ben Kieffer és társai bemutattak néhány kísérleti és elméleti eredményt, amelyek szerint a robot pontossága függ a pályapontok munkatérbeli elhelyezkedésétől. A munkadarab legkedvezőbb elhelyezését támogató módszert a munkájuk során nem sikerült teljesen kidolgozniuk [18].

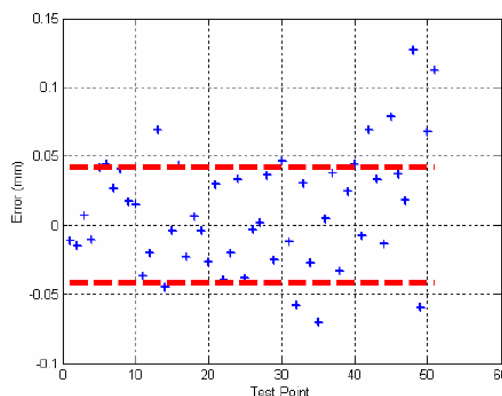
1999-ben Hrman és társai kifejlesztettek egy módszert, ami úgy tolja el pályapontok helyzetét a munkatéren belül, hogy a robot a legmerevebb konfigurációjában érje el azokat [19].

2003-ban Veryha és Kurek tovább fejlesztették Hrmanék módszerét. Az előző módszer nagy hátránya volt, hogy eltolta a pályapontokat, ezáltal a munkadarabot is újra kellett pozicionálni. Veryha-ék kifejlesztettek egy algoritmust, amely egy adott pályapontban megkeresi azokat a csukló szögelfordulásokat, ahol a helyzetpontosság hibája nem haladhatja meg a megadott határértéket. Ennél a módszernél a pályapontok helyzete nem változik, csak a pontba vezető mozgást kell újratervezni az új konfigurációval. A módszer segítségével sikerült az ipari robot pontatlanságát 3%-ról sikerült 1%-ra csökkenteni. Továbbá hozzájárult, hogy az iparban a robottal mért alkatrészek száma 5%-ról 15%-ra növekedjen [20].

2005-ben Hui és társai az ipari robotok merevségének meghatározásával kapcsolatos cikket prezentáltak. Munkájuknak két fő témája volt: növelni a robotos megmunkálás minőségét valamint valós idejű optimalással maximalizálni a leválasztott anyagmennyiséget [21].

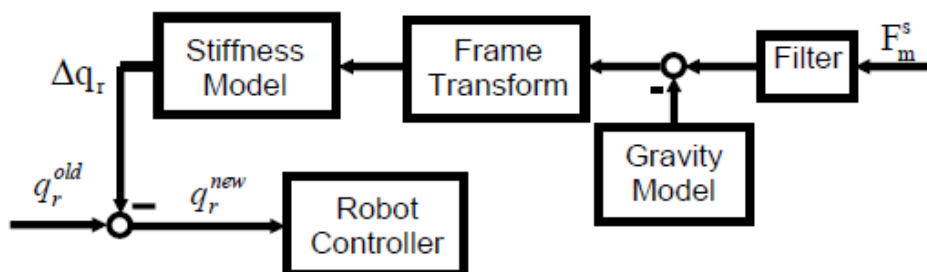
Az első esetben a pontosság növelését a forgácsolási erők okozta deformáció kompenzálásával akarták megvalósítani. Elképzelésük az volt, hogy egy elegendően pontos modell készítsenek robot merevségéről, ami alapján kiszámolható a várható deformáció. Másrészt a módszer legyen kellően egyszerű, hogy valós idejű beavatkozást tegyen lehetővé. Ha a szerkezet összes mechanikai elemét beleveszik a modellezésbe, a számítási igény túl nagy lett volna, ami képtelenné teszi a valós idejű kompenzálást, valamint a paraméterek meghatározási is nehézséget okozott volna. Mivel a robotkar elemeit a gyártók a lehető legmerevebbre igyekeznek legyártani, jelen esetben deformálódásuk mértékét elhanyagolták. Ebből kifolyólag a nagy eltérések fő oka a csuklók engedékenysége, amit az erőátviteli elemek rugalmassága okoz. Így a robot merevségi modelljét a 6 csukló merevségére csökkentették le. Ennek eredménye, hogy a deformációk közvetlenül kompenzálhatók, ha a csuklók elfordulás szögeit referencia jelként adják a szervók vezérlésébe. A modell meghatározásához a robotra egy gömbvégű szerszámot szereltek. A szerszámot 100-400N közötti erővel terhelték, aminek hatására a szerszám elmozdult. Az eredeti és az elmozdult állapot közötti eltérést ROMER koordináta mérőgéppel mérték meg. A mérést számos orientációban és pontban is elvégezték. Minden mérésnél kiszámolták a deformációkat a három Descartes koordináta irányában. Ezek alapján kiszámolták a legkisebb négyzetek módszerével közelítve a merevségi mátrixot. A mérési eredményeket a 7. ábra

mutatja. Az különböző pontok elmozdulás értékeinek többsége jól láthatóan egy szűk sávban helyezkedik el. Ezért a modellezéshez állandó paramétereket használtak [21].



7.ábra: Különböző pontokban mért elmozdulás értékek [21]

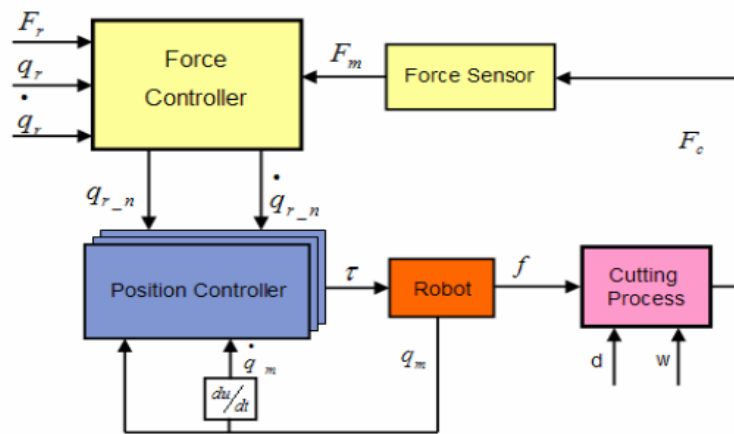
A kompenzálás blokk diagramja a 8. ábrán látható. Az erőmérőből folyamatosan érkeznek a mért forgácsolási erőértékek. A zajmentesítés után, a főorsóra ható nehézségi erőt kivonták. A tiszta forgácsolási erőket robot koordináta rendszerbe kellett átranzformálni. A merevségi modell alapján számolt, az adott erő okozta deformáció nagysága szolgált hibajelként. Az elméleti pozíció és a deformáció különbsége képezte a robotvezérlés bemenő jelét [21].



8. ábra: A valós idejű deformáció kompenzálás blokkdiagramja [21]

A második esetben a forgácsleválasztási teljesítmény maximalizálása volt a cél. A lényeg, hogy leválasztott anyagmennyiséget az egész megmunkálás alatt állandó értéken tartsák. Ezt, az addigi gyakorlattól eltérően, az előtolás folyamatos változtatásával próbálták elérni. Ezáltal a fogásmélység és fogásszélesség mellett az előtolás is mindig maximális értéket vett fel. A folyamatot valahogy szabályozni kellett. Közvetlenül leválasztott anyagmennyiség mérését nagyon nehéz lett volna megvalósítani. Ezért inkább forgácsolási erő mérésén keresztül szabályozták az előtolást. Az erőt egy a robotkarimája és a szerszám közé szerelt, 6 szabadságfokú erőmérőcellával mérték. A megengedett erő értékét a főorsó

teljesítménye szabta meg. Az állandó forgácsolási erő előnye, hogy a várható szerszám éltartam is jól meghatározható. A folyamat szabályzóköre a 9. ábrán látható. A rendszer tartalmazza az előzőleg megvalósított deformáció kompenzálást is [21].



9.ábra: Erőszabályzás a robotos megmunkáláshoz [21]

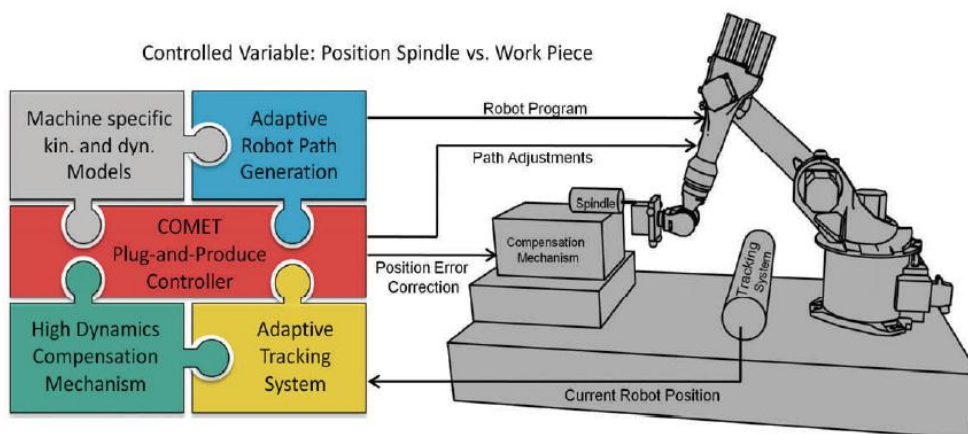
A gyakorlati kísérleteket öntött alumínium munkadarabokon végezték. Az erőszabályozás használatával a megmunkálási idők 30-50%-kal csökkentek munkadarabtól függően. A megmunkált felület mérésére lézeres elmozdulás érzékelőt használtak. A kompenzált forgácsolással kapott munkadarab felületi hullámossága 0,1 mm-en belül volt, ami nagyságrendileg megegyezik a robot pontosságával. A tesztek során hengeralak megmunkálását is vizsgálták. A kompenzálás előtti 0,9 mm-es hibát az erőszabályzó alkalmazásával sikerült 0,3 mm-re csökkenteni. Ez lehetővé teszi, hogy előmegmunkálási feladatokat, melyek pontossága nem szokta meghaladni a 0,5 mm-t, ipari robotok is ellássanak [21].

2010-ben Vosniakos és Matsas publikált egy cikket, amelyben a forgácsolási erők okozta hatások csillapításával foglalkoztak. Ők két szempont alapján végezték a kutatásukat. Az egyik, hogy a munkadarab a robot munkaterében azt a helyet foglalja el, ahol a robot statikus és dinamikus tulajdonsága is a legkedvezőbb marási feladatokhoz. A második szempont pedig, hogy a megmunkálásnak megtalálják azt a kiindulási pontját és robotkonfigurációját, amiből indulva a csuklóban a legkisebb nyomatékok ébrednek. A megoldás meghatározásához nagyon sok esetet kellett megvizsgálni. Ez analitikus módszerrel nagyon nagy számítási igényt jelentett volna. Emiatt inkább genetikussal keresték, a legkedvezőbb pozíciókat. Mind a két szemponthoz külön algoritmust írtak. Az egyedeket 6 tengelyes, humanoid robotra generálták. A módszer hatékonysága abban rejlett, hogy az eddigi kutatásoktól eltérően, kinematikai manipulabilitás mellett a dinamikai is figyelembe lett véve [22].

2011-ben Andres és társai egy külső lineáris tengellyel ellátott, 6 tengelyes ipari robottal végeztek kísérleteket. A munkadarab egy vízszintes forgó asztalra volt felfogatva, így összesen 8 tengely szimultán vezérlését kellett megoldani. A robot munkaterét bővítő külső tengelyek hátránya, hogy a rontják pontosságot. Andres-ék célja a teljes robotcella kalibrálása volt. Munkájuk során egy robotkarimára szerelhető lézeres elmozdulás érzékelőt használtak. Kifejlesztették az ún.: érintkezésmentes rögzített síkbeli kalibrációs folyamatot. A módszer előnye, hogy viszonylagosan olcsó, és könnyen automatizálható. A módszer segítségével a kalibrálás előtti átlagos eredő hiba 5,429 mm-ről 0,998 mm-re csökkent [23].

2.2. COMET

2010 szeptemberében elindult a COMET (COmponents and METhods) elnevezésű 30 hónapos, Európai Unió projekt. A kutatásban 8 ország 14 kutatócsoportja vett részt. A projekt célja egy módszer kidolgozása, amely magába foglalja a robotok kinematikai és dinamikai modelljét, adaptív pályakövető rendszert, High Dynamics Compensation Mechanism-t (HDCM), lehetővé téve az adaptív pályagenerálást (10. ábra) [3].



10. ábra: COMET-COmponents and MEthods [3]

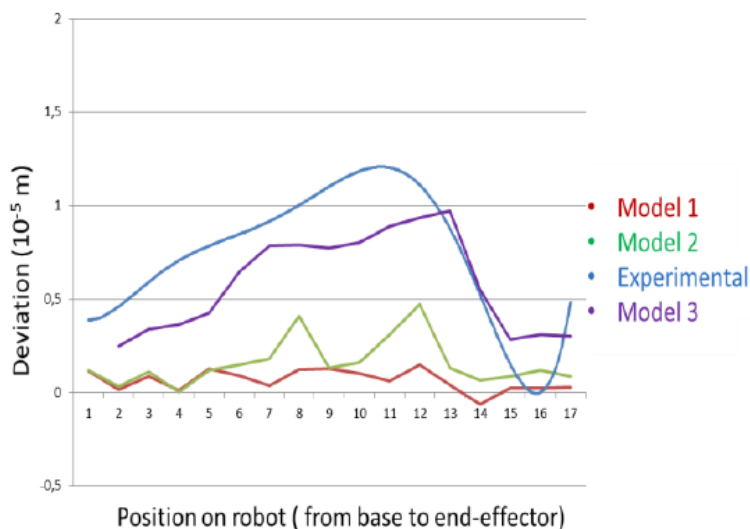
A következőkben néhány, a COMET-en belüli kutatási eredményt fogunk ismertetni.

2.2.1. Dinamikai modell létrehozása

Már korábban is láthattuk, hogy az ipari robotok esetében a pontos megmunkálást a deformációk hatékony kompenzálásával lehet elérni. A megfelelő kompenzálás alapja egy valóságghű dinamikai modell.

A következőkben ismertetett kutatás során az ipari robot szerkezeti viselkedését véges elemes környezetben vizsgálták, és határozták meg a deformációkat okozó főbb tényezőket. A számított eredményeket gyakorlati tesztekkel vetették össze, és ez alapján finomították a virtuális modellt [24].

STEP formátumú CAD modellt használtak a véges elemes számításokhoz. A modelleken több módosítást is végre hajtottak, hogy a hálózás megfelelő minőségű legyen. Viszont figyelni kellett, hogy a változtatások ne menjenek a pontosság rovására. A kényszereket az elemek között a véges elemes környezetben definiálták. Ez lehetővé tette a robot statikus helyzetének változtatását, így több különböző orientációban is elvégezheték a szimulációt. Három féle modellel dolgoztak. Az első volt az alap modell, ennek a tovább fejlesztett verziója a többi kettő. Különbségek a csatlakozások definiálásánál adódtak. Az alap esetben az elemek közötti csatlakozás fixnek tekintett. Tehát csak a karelemek rugalmassága okozza a deformációt. A második eset is fix csatlakozásokkal dolgozik, csak a csatlakozó felületeken végeztek módosításokat. A harmadik modellnél a pontosság sokat javult, köszönhetően, hogy a csuklók korlátozott merevsége is figyelembe lett véve egy meghatározott csavaró merevséggel. A 11. ábrán a három féle modell pontossága látható [24].

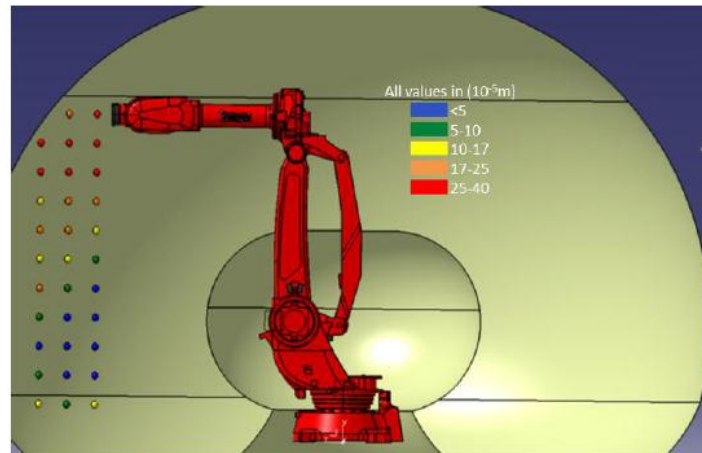


11.ábra: Véges elemes számítások összehasonlítása a gyakorlati eredményekkel [24].

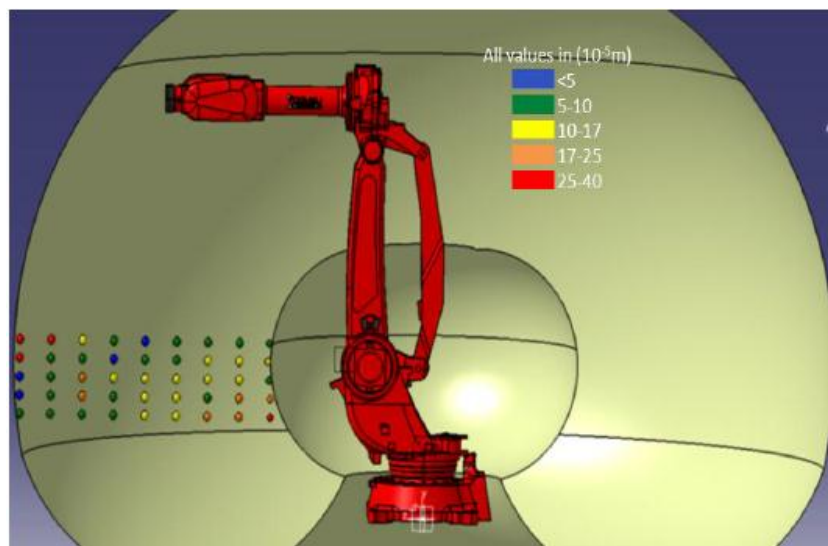
A gyakorlatban mért eredmények is alátámasztották, hogy a karelemek rugalmassága elhanyagolható a csuklók engedékenységehez képest. Ezen belül is az első 3 csukló az, ami a legjobban hozzájárul a keletkező deformációkhoz. A harmadik típusú modell pontosságát iterációs úton lehetett még tovább javítani [24].

A pontosan meghatározott dinamikai modell alapján a robotmerevségének feltérképezése a munkaterének különböző pontjain egyszerűen megvalósítható volt már. A szerzők elkészítették a robot munkaterének ún.: elmozdulás térképét (12. ábra). A jobb szemléltetés érdekében a deformáció mértékét öt szintbe sorolták be, melyeket különböző színekkel jelöltek. Külön vizsgálták a munkatér vízszintes és függőleges kiterjedésének deformációit [24].

a)



b)



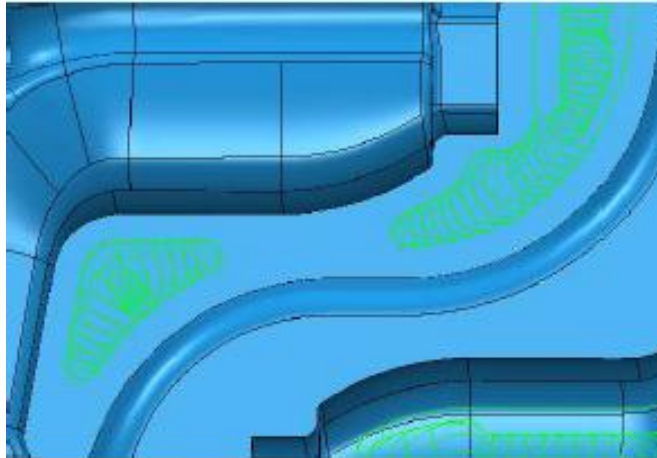
12.ábra: a) Függőleges irányú elmozdulás térkép b) Vízszintes irányú elmozdulás térkép [24]

2.2.2. Deformációk kompenzálása I.

A projekt alatt több különböző módszer lett kifejlesztve a robotok megmunkálási hibáinak csökkentésére. Lehmann és társai egy 3 lépésből álló mechanizmust fejlesztettek ki. Az első két lépés off-line, míg az utolsó online kompenzálást tartalmaz [25].

Első lépés a megfelelő megmunkálási stratégia létrehozása. Fontos szempont a munkadarab elhelyezése. Ha van mód rá, elkerülendők nagy tengelyelfordulások és olyan helyzetek, mikor nincs valamelyik tengelyen terhelés, mivel ezek holtjátékot okozhatnak a csuklóknál. Lényeges pont a folyamat során ébredő erők csökkentése. Ipari robotok esetében az előtolásra merőleges irányban ez különösen igaz. Ebben az irányban ébredő erő, ellenirányú marással és körülbelül a szerszámtárcsa felét átfogó kontakt szöggel minimálisra

csökkenthető. Toroid mozgásokból álló szerszámpályát alkalmazva a kontakt szög állandó értéken tartható (13. ábra) [25].



15.ábra: Szerszámpálya trochoid mozgásai [25]

Az optimális szerszámpálya létrehozása után biztosítottak a stabil megmunkálás alapjai. A következő lépés a forgácsolási erők generálta eltérések off-line szintű kompenzációja. Ha rendelkezésre állnak a munkadarab és szerszám anyagpárosítására vonatkozó anyagjellemzők, a szerszámpálya generálása után a forgácsolási paraméterek figyelembe vételével minden egyes pontban kiszámíthatóak a forgácsolási erők. Ha ismert a robot merevségi modellje, az erők kiszámítása után becsülhető a robot szerkezetének deformációja. A már létező szerszámpálya a szükséges korrigálások beillesztésével frissíthető [25].

Csak off-line kompenzáció alkalmazásával egy bizonyos szintig javítható a megmunkálási pontosság, de a nagyfrekvenciás rezgések által okozott vagy az előre nem meghatározható zavarásokat nem lehet ilyen módon korrigálni. (Nagy frekvencia alatt, azt az értéket értjük, ahol a periódus idő a robotvezérlő ciklusidejétől kisebb.) Ezek kiküszöbölését, a szerzők további online korrekcióval oldották meg [25].

A tényleges forgácsolási erőket a robot szerszámkarimája és az adapter tárcsa közé szerelt erőmérő szenzorral határozták meg. A 16. ábrán az elméleti és a tényleges erőértékek láthatóak. Az off-line módon számított erőket összehasonlítva a valóságban mért erőkkal, egy külső eszköz kiszámolta a valódi deformáció és számolt deformáció közti eltérés. Az adott ponthoz tartozó eltérést a robotvezérlőbe vezetve történik a pályapont korrekciója. Ha az erők aktuális pontjain kívül a következő néhány pont adatait is elküldi a controllerbe, közelítő interpoláció vagy előre tekintő funkciók is megvalósíthatók [25].



16. ábra: Az off-line módon számított- és az erőmérőcellával mért forgácsolási erőértékek [25]

Az offline módon számított erőértékek 15-20%-os pontossággal közelítették a valóságban mért értékeket. A gyakorlati kísérletek során a módszer alkalmazásával a munkadarabok pontosságának 30%-os javulását is el tudták érni [25].

2.2.2. Deformációk kompenzálása II.

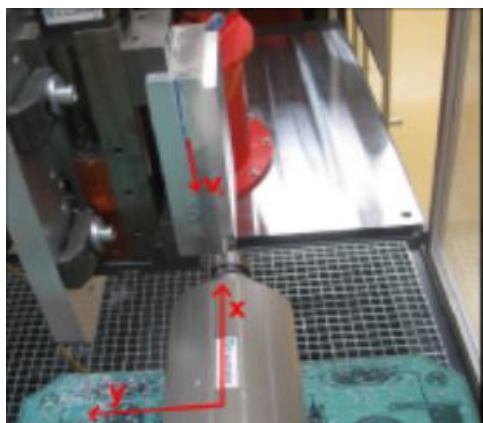
A COMET-en belüli EU/FP7 projektnek is a robotos marás pontosságának javítása volt a fő célja. A nagy pontosságú megmunkálás megvalósításához egy HDCM (High-Dynamics Compensation Mechanism) 3 dimenziós piezo aktuátor működtetésű kompenzáló mechanizmust alkalmaztak, ami már egy kifejlesztett technológia volt ekkor. A kísérletnél a robot mozgatta a munkadarabot és a marószerszámmal ellátott főorsó különálló egységet képezet. A kompenzáló a szerszámnál alkalmazták [26].

A gyakorlati megvalósításnál az alumínium (AlMg3,5) hasáb munkadarabot REIS RV40 típusú ipari robotot mozgatta. A főorsó egy mikro manipulátorhoz lett csatlakoztatva, amely lehetővé tette a főorsó mozgását x, y és z irányban. Helyzetét kapacitív szenzorral mérték. A piezo aktuátort terhelés mérővel látták el. A kompenzáló mechanizmus egy dSPACE típusú DS1103 vezérlő asztalon keresztül csatlakozik a mikromanipulátorhoz. A marás során létrejött eltérések mérését (aminek a kompenzálása a mikromanipulátor feladata) egy KEYENCE típusú LK-G87 lézeres szenzorral valósították meg, melynek pontossága 0.2 mikrométer. A megmunkálás során keletkező forgács keresztezhetette a lézernyaláb útját,

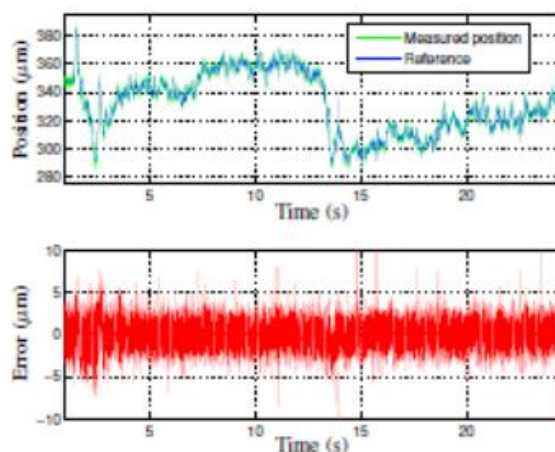
pontatlanná téve a mérést. Ezért fontos volt a hatékony zavorszűrés megvalósítása. A mért eltérések referencia jelként lettek bevezetve a micromanipulátor szabályzó körébe [26].

A kísérleteknél egy sík- és kettő palástmarási esetet vizsgáltak. Síkmarást a mikromanipulátor x- irányában hajtották végre. A kísérlet elrendezését és eredményeit a 17. a és b ábra mutatja. A 17.b ábrán jól látható, hogy az eltérések $\pm 7\mu\text{m}$ belül maradtak [26].

a)

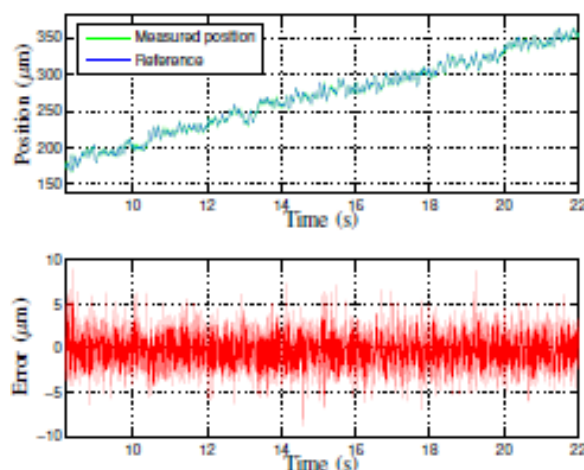
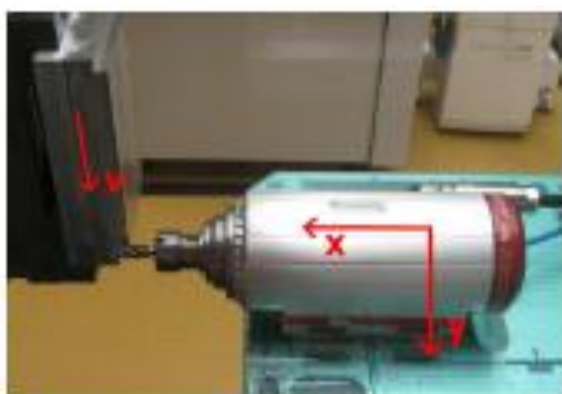


b)



17. ábra: a) Kísérleti elrendezés b): Referencia érték és a kapacitív szenzor által mért pozíció, a micromanipulátor x irányában történő síkmarásakor (felül) és a hozzá tartozó hibajelek (alul) [26]

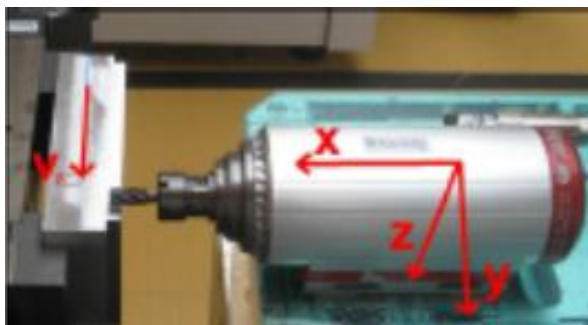
Az első palástmarási kísérletet a mikromanipulátor y - irányában végezték el. Az elrendezést és az eredményeket a 18. a és b ábra mutatja. A 18. b ábrán alapján megállapítható, hogy a síkmarási esethez hasonlóan itt is $\pm 7\mu\text{m}$ között voltak az eltérések [26].



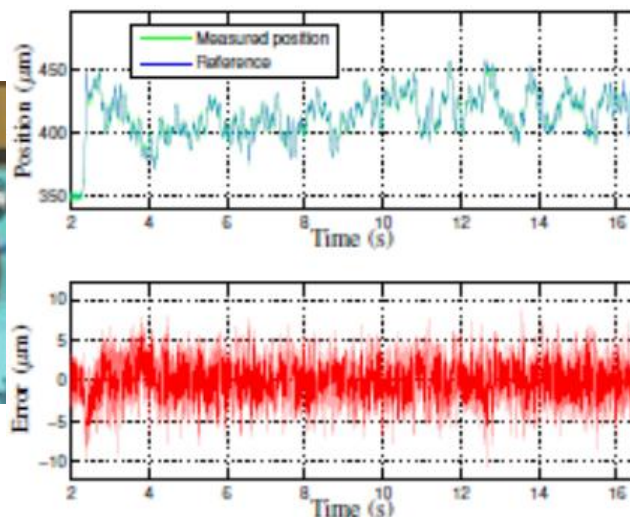
18. ábra: a) Kísérleti elrendezés b): Referencia érték és a kapacitív szenzor által mért pozíció, a mikromanipulátor y - irányában történő palástmarásakor (felül) és a hozzá tartozó hibajelek (alul)[26]

A második palástmarási kísérletet a mikromanipulátor z irányában végezték el. Az elrendezést és az eredményeket a 12. a és b ábra mutatja. A 12. b ábrán látható, hogy a ennél az elrendezésnél adódtak a legnagyobb eltérések, ami $\pm 12 \mu\text{m}$ -t jelent [26].

a)



b)



19 ábra: a) Kísérleti elrendezés b): Referencia érték és a kapacitív szenzor által mért pozíció, a mikromanipulátor z - irányában történő palástmarásakor (felül) és a hozzá tartozó hibajelek (alul) [26]

A módszer hatékonyságának teszteléséhez a kísérletet elvégezték a mikromanipulátor használata nélkül is. A kompenzált és nem-kompenzált eset összehasonlítása az 1. és 2. táblázatban látható. A táblázatok értékei is jól szemléltetik, hogy a mikromanipulátor alkalmazásával az átlagos eltérések mind a 3 elrendezésnél, több mint a felére csökkentek [26].

TABLE I
MAXIMUM ERROR (e_m) OF THE MILLING PROFILE

Axis	e_m compensated (μm)	e_m uncompensated (μm)	Ratio
X	13.99	18.34	1.31
Y	12.83	29.25	2.28
Z	24.52	66.99	2.73

1. táblázat: A kompenzált és a nem-kompenzált marás során előfordult maximális eltéréseket [26]

TABLE II
STANDARD DEVIATION (σ_e) OF THE MILLING PROFILE

Axis	σ_e compensated (μm)	σ_e uncompensated (μm)	Ratio
X	2.80	7.57	2.71
Y	2.54	5.55	2.18
Z	4.68	14.93	3.19

2. táblázat: A kompenzált és a nem-kompenzált eset átlagos eltérései [26]

3. Munkakörnyezet bemutatása

Ebben a fejezetben a dolgozat elkészítése során használt eszközöket, és szoftvereket mutatjuk be.

3.1. Kísérleti elrendezés

Robotos megmunkálást a Gyártástudomány és -technológia Tanszék robotlaborjában található két ABB robottal végeztük (1. ábra). A robotok típusa: IRB140, és IRB1400h. S4C+ típusú vezérlővel.



18. ábra: Robotcella

A megmunkálásokhoz szükség van egy, a robot karimájára szerelt megmunkáló szerszámmra. Erre szintén a tanszéken található pneumatikus főorsót használtunk. Fordulatszama: 5000 1/perc. A szerszám befogása szorító patronnal történik, melynek belső átmérője 6 mm. Mivel a főorsó és a robotkarima lyukköre nem volt azonos átmérőn, így felszereléshez terveztünk egy egyszerű adapter tárcsát. Az adapter tárcsa vastagságának csökkentése érdekében a két lyukkört egymáshoz képest 45°-al elforgattuk. (Lásd: Melléklet)

A szerszám bemérése céljából beszerelésre került az adapter tárcsa, és a robotkarima közé egy JR3 típusú hattengelyes erőmérő szenzor, amelynek segítségével elvégezhető a szerszám és munkadarab bemérése is.

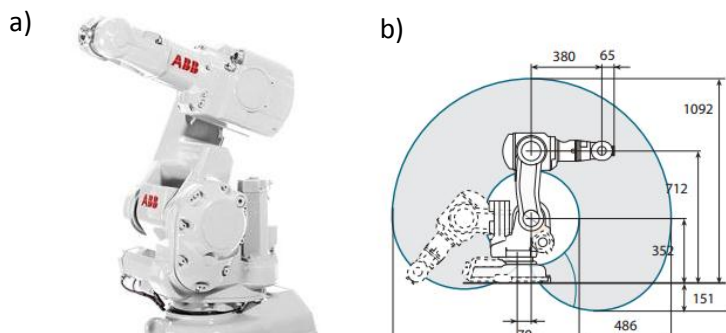
Megmunkálások során a porelszívást/porelvezetést manuálisan oldottuk meg egy porszívó segítségével.

3.1.1. ABB IRB 140

Az ABB kisméretű robotkarja.

Teherbírása: 6 [kg]

Maximális kinyúlás: 810 [mm]



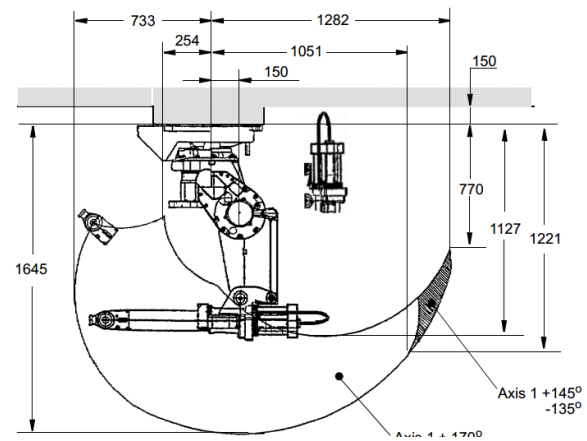
19. ábra: a) IRB 140 b) a robot főbb geometriai méretei [27]

3.1.2. ABB IRB 1400h

Teherbírása: 5 [kg]

Maximális kinyúlás: 1440 [mm]

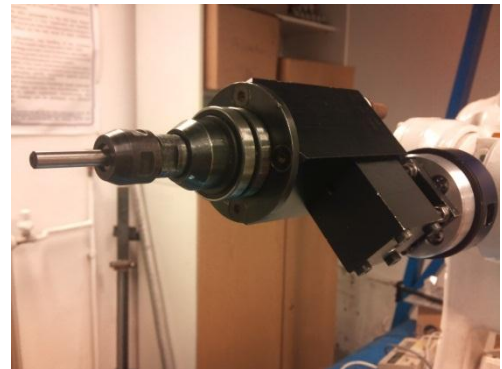
Legnagyobb különbség az IRB140-hez képest a kinematikai felépítése. Alapvetően soros kinematikájú ez is, de tartalmaz egy zárt láncot. Ezen tulajdonságának kezelése nehézséget okoz egyes nem robotos eredetű CAM szoftvereknek (Lásd 4.2. fejezet!).



20. ábra: ABB IRB 1400h [28]

3.1.3. Főorsó, és szerszám

A maráshoz használt pneumatikus főorsó a 21. ábrán látható. A főorsó működtetése egy pneumatikus útszelepen keresztül történt, melyet a robotvezérlő digitális kimeneten keresztül vezérelt.



21. ábra: Pneumatikus főorsó

3.1.4. JR3 erőmérő szenzor

JR3 egy 6 tengelyes erőmérő szenzor, melynek segítségével mind a hat szabadságfok irányában mérhetők az erő, illetve nyomaték értékek. Kialakításának köszönhetően a robotra egyszerűen felszerelhető. Jelét (közvetett módon) visszacsatolva a robotvezérlőbe automatikus szerszám bemérést valósíthatunk meg vele.

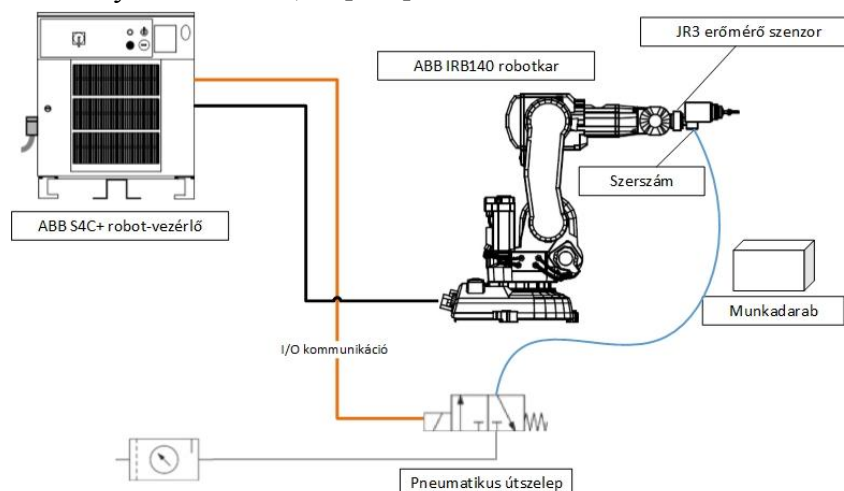
Felbontása:

Erő: 0,1 [N]

Nyomaték: 0,01 [Nm]



22. ábra: JR3 szenzor [29]



23. ábra: Robotcella architektúrája

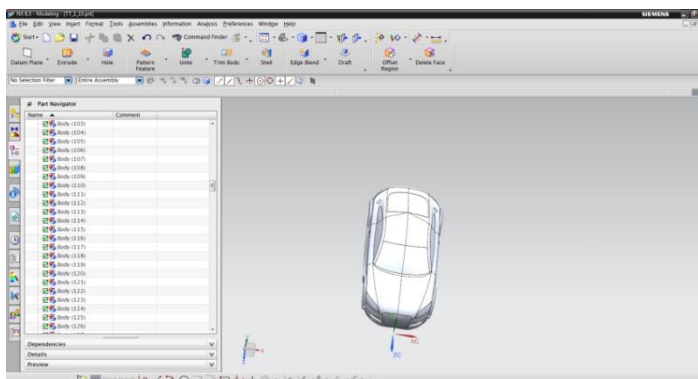
3.2. Virtuális munkakörnyezet, szoftverek

A robotok mozgáspályáinak létrehozására számos módszer létezik. Egy marási művelet esetében a szerszámpályája több ezer (akár több tízezer) pályapontot is tartalmazhat. Így a hagyományos betanításos módszer itt nem igazán alkalmazható. Ezért célszerű egy CAM szoftver segítségével elvégezni a pályagenerálást. Mint már a szakirodalmi áttekintésben is említettük, egy általános CAM szoftver nem elegendő a robot programozásához. A kutatási kérdésünk, hogy az alapvetően robotos gyökerekkel rendelkező robotszimulációs rendszerek, vagy az alapvetően szerszámgépes gyökerekkel rendelkező CAM szoftverekkel lehet a robotprogramot előállítani, az ABB robotok esetében a RAPID programot.

Ennek megfelelően 2 módszert vizsgáltunk a megmunkálások robotprogramjainak elkészítésére. Első esetben egy különálló CAM szoftverben készítettük el a szerszámpályákat (Siemens NX 8.5), majd a létrehozott pályát egy offline robotprogramozó (Siemens Tecnomatix RobotExpert, vagy Process Simulate) szoftverben szimuláltuk, és posztprocesszáltuk. Mindkét szoftverhez a Gyártástudomány és –technológia Tanszéktől kaptunk liszenszet. A másik módszernél a Delcam PowerMill CAM szoftverét használtuk, amelyhez elérhető egy robotos kiegészítő csomag. Így egy szoftverben történik a szerszámpálya generálás és a robotprogramozás is. A szoftverhez a Tanszéken keresztül a Delcam Hungary Kft-től kaptunk 6 hónapos liszensz jogot.

3.2.1. Siemens NX 8.5

A Siemens integrált megoldást nyújtó tervezőrendszere. Egy rendszeren belül lehet a tervezést, megmunkálást, analízist és rajzokat elkészíteni, amelyből dolgozatunk során a modellezési, illetve a megmunkálási funkciókat használtuk.

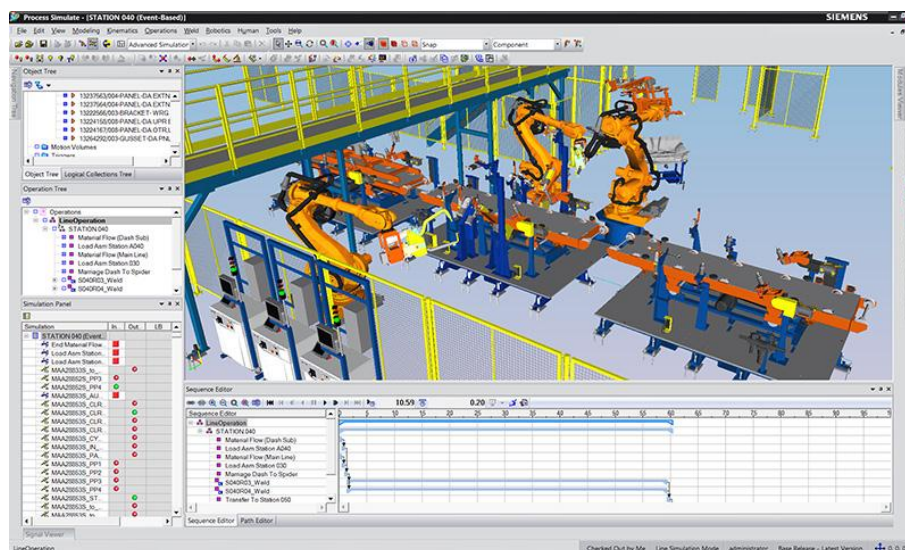


24. ábra: Siemens NX 8.5

3.2.2. Siemens Tecnomatix RobotExpert / Process Simulate

A Process Simulate egy digitális gyártást megvalósító szoftver. Segítségével, a robottal végezhető legtöbb feladat off-line módon programozható. Virtuális beüzemelés tesz lehetővé teljes gyártási folyamatokra. A robotprogramozáson kívül, a robotok környezetében lévő PLC-k működése is szimulálható.

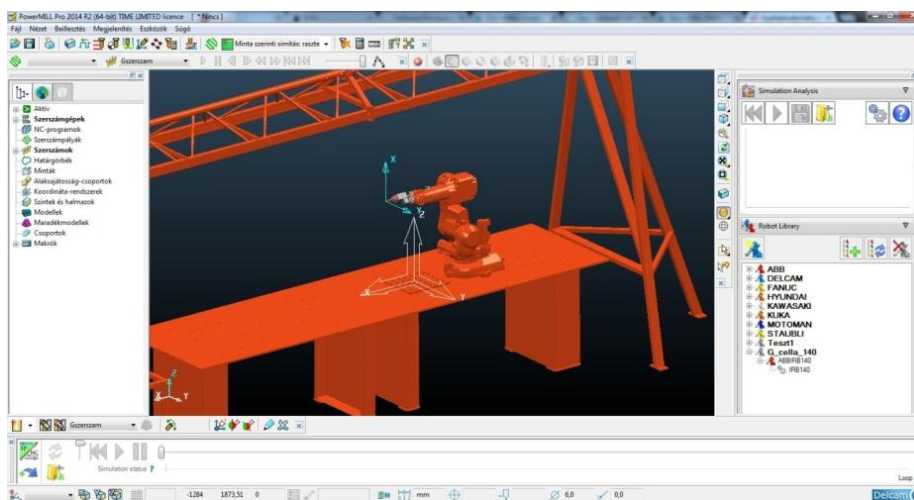
RobotExpert csupán annyiban tér el az előzőtől, hogy annak „csak” a robotalkalmazás részét tartalmazza.



25. ábra: Process Simulate [30]

Elérhető hozzá ún. „CLS Upload” funkció, amelynek használatával NX szoftverben létrehozott szerszámpályákat értelmezni képes.

3.2.3. Delcam PowerMill Robot Interface



26. ábra: Delcam PowerMill Robot Interface

A Delcam PowerMill egy öttengelyes CAM szoftver. Telepíthető hozzá Robot Interface, amely lehetőséget ad szerszámpályát robotok részére is előállítani. A kiegészítő inkább csak megmunkáló, esetleg hegesztő vagy ragasztó feladatok programozására alkalmas. Megoldható mozgatott vagy fix főorsójú marási műveletek programozása is. Utóbbinál a robot mozgatja a munkadarabot, míg a főorsó és a szerszám fix helyzetben van. [31]

4. Robotok programozása marási műveletekre

4.1. Siemens NX 8.5 & Tecnomatix Process Simulate esetében

NX, és PS esetében az alábbi fő lépésekre bonthatjuk egy marási művelet robotprogramjának elkészítését:

1. Szerszám pályák elkészítése NX rendszerben, majd annak CLS fájlba való kiírása
2. Process Simulate-ben a robot, és cella betöltése
3. CLS fájl beolvasása a programba, szerszám orientációjának beállítása
4. Munkadarab elhelyezése a robot munkaterében
5. Szimuláció
6. Posztprocesszálas RAPID program nyelvre
7. Program feltöltése a robotra, majd futtatása

1) Szerszám pályák elkészítése NX rendszerben

Első lépésben CAD modell alapján elkészítjük a marás szerszámpályáját NX CAM szoftverben.

2) Process Simulate-ben a robot, és cella betöltése

A cella összeállításhoz szükséges az cellában található eszközök CAD modellje. Egy előzetesen telepített CAD fordítóval a szoftver már a kiterjesztések többségét kezelni képes. Továbbá lehetőség van RobCAD-ben összeállított cella átkonvertálására is. Robot esetében betöltéséhez szintén a legegyszerűbb megoldás egy RobCAD modell átkonvertálása, mivel ezek már rendelkeznek a robot kinematikai leírásával, de akár új robot kinematikai felépítésére is van lehetőség.

3) CLS fájl beolvasása a programba, szerszám orientáció beállítása

A két szoftver között a szerszámpálya átvitelére „cls” fájl kínál megoldást, mivel ezt képes kezelni mindkét szoftver.

```
24 GOTO/32.6951,49.3572,0.0000
25 GOTO/32.7015,49.0861,0.0000
26 GOTO/32.6235,48.9732,0.0000
27 GOTO/31.8960,48.0736,0.0000
28 GOTO/30.9843,47.1829,0.0000
29 GOTO/29.8885,46.3021,0.0000
30 GOTO/28.6314,45.4613,0.0000
31 GOTO/27.3231,44.8008,0.0000
32 CIRCLE/23.4897,53.7468,0.0000,0.0000000,0.0000000,1.0000000,9.7327,0.0300,0.5000,8.0000,0.0000
33 GOTO/20.4277,44.5083,0.0000
34 GOTO/19.0722,45.0582,0.0000
35 CIRCLE/23.2370,53.7019,0.0000,0.0000000,0.0000000,1.0000000,9.5947,0.0300,0.5000,8.0000,0.0000
36 GOTO/15.4996,48.0281,0.0000
37 GOTO/14.5493,49.4802,0.0000
38 GOTO/13.7237,51.1606,0.0000
39 GOTO/13.0340,53.0582,0.0000
40 GOTO/12.4811,55.1737,0.0000
41 GOTO/12.0649,57.5087,0.0000
```

27. ábra: CLS fájl adatstruktúra

Process Simulate „CLS Upload” funkciójában a szerszámorientáció beállítására a következő lehetőségek vannak:

- Fixed: Egy a programban kiválasztott referencia koordináta rendszernek megfelelően állítja be a szerszám orientációt.
- Tangent: A programba betöltött munkadarab modell felületéhez képest tangenciálisan veszi fel az irányt. Tangenciális beállításnál a TCP koordinátarendszer X tengelyét a pályairányára érintőlegesen állítja.
- Toward point: Mindig egy adott fix pont felé irányítódik a szerszám
- Tangent ZigZag: Tangenciálishoz hasonlóan, ha ZigZag stratégiával lett elkészítve a szerszámpálya.

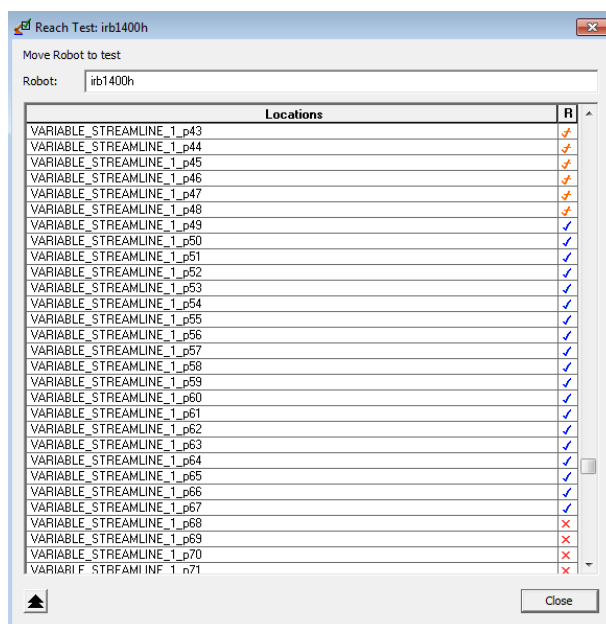
Mivel az orientáció beállítására csak az előzőleg említett négy lehetőség van, ezért ez jelentősen lekorlátozza az alkalmazási lehetőségeket. Bonyolult geometriák megmunkálása gyakran speciális szerszámorientációt igényel, amelyekre nem elegendő a fent kínált négy beállítás. Ilyen lehet például 5 tengelyes palástmarás, vagy gömbmaró változó vezető/elhajló szög esetén, amelynek ütközések elkerülésében van nagy szerepe.

A következő lépés a cls fájl betöltése a szoftverbe. Itt újabb problémába ütköztünk: Egy-egy szerszámpálya viszonylag egyszerűbb geometria esetében is gyakran több ezer pontból állhat. Ekkora pontthalmazt a Process Simulate már nem tud feldolgozni (érthető is, hiszen nem erre lett kifejlesztve), így ennek a módszernek az alkalmazhatósága nagyban függ a szerszámpálya méretétől.

4) Munkadarab elhelyezése a robot munkatérében

A robotok munkatere úgynevezett irányos munkatér, vagyis a munkatéren belül az egyes pontok nem érhetőek el a robot számára minden irányból. A munkatér szélét közelítve egyre csökken a lehetőségek száma, míg a határon már csak egyetlen irányból érhető el az adott pont. Ezért a munkadarab elhelyezésének szoftveres támogatása nagy segítséget jelent a megfelelő munkadarab elhelyezkedés megkeresésénél (főleg nagyobb munkadarabok esetében)

Ehhez RobotExpert/Process Simulate rendelkezik egy „Reach Test” nevű funkcióval (28. ábra), amelyet kiválasztva kijelölhetjük az összes pálya elemet (pontot). Majd a munkadarabot, és a pályát mozgatva a térben folyamatosan jelzi, mely pontok érhetőek el (28. ábra: kék pipa), melyek nem (28. ábra: piros X), és melyek csak konfiguráció váltás árán (28. ábra: áthúzott pipa). Így könnyen megkereshető az a helyzet – ha egyáltalán létezik olyan – ahol a robot képes bejárni a teljes szerszámpályát.



28. ábra: Process Simulate „Reach Test”

5) Szimuláció:

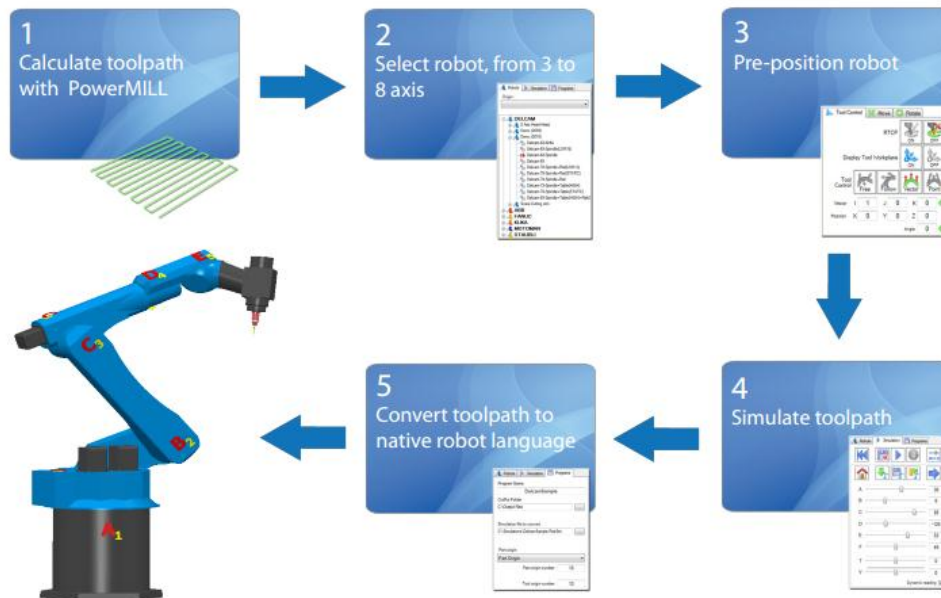
Minden robotprogramozást végző szoftver egyik alap funkciója a szimuláció. Ezt használva ellenőrizhetjük a programozott műveletet:

Egyrészt ütközésvizsgálatot végezhetünk: Mivel a szerszámpálya generálás, illetve a robot kezelése 2 külön szoftverbe történik, ezért robotos szimuláció esetében már a megmunkálás szimulációját nem tudjuk elvégezni. Így az előgyártmánnyal való ütközés nehezen szűrhető ki.

Másrészt a konfiguráció váltásokat figyelembe kell venni: A robotok munkatere mindig a konfigurációs alterek összege, az alterek a munkatér központi tartományaiban átfedik egymást. A munkatér szélein viszont csak egyetlen egy konfigurációban tud a robot mozogni. Illetve a munkatér belüli helyzetek sem érhetőek el tetszőleges konfigurációban. Optimális, ha egy szerszámpályát a robot konfiguráció váltások nélkül tud befutni. Ha ez nem kivitelezhető, akkor ezeket ellenőrizni kell, hogy ne ütközzön egy-egy ilyen váltás során se az aktuális munkadarabnak, se a cellában található többi tárgynak. Ezekre megoldás lehet egy-egy köztes „via” pont beszúrása, mellyel biztonságosan végbe tud menni a váltás.

4.2.Delcam PowerMill

PowerMill esetében egyszerűsödik a folyamat, mivel egy szoftver fedi le a szerszámpálya generálást, és a robotprogramozási részt.



29. ábra: Marási műveletek robotprogramozásnak folyamata Delcam esetében [32]

1. Szerszám pályák elkészítése
2. Robot és cella környezetének betöltése
3. Munkadarab elhelyezése a robot munkaterében
4. Szimuláció, szerszámpálya ellenőrzés
5. Posztprocesszálas RAPID program nyelvre
6. Program feltöltése a robotra, majd futtatása

1) Szerszám pályák elkészítése

Itt is az első a betöltött CAD modell alapján a szerszámpályák elkészítése, amelyek ugyanúgy történnek, mint egy 5 tengelyes szerszámgép esetében.

2) Robotcella betöltés, összeállítás

A robotcella betöltéséhez először azonban azt összeállítani szükséges. PowerMill esetében ezt ún. „mtd” fájlban készíthetjük el, amely egy xml struktúrájú fájl.

```

<machine_part NAME="RobotAxis2">
  <axis>
    <control_info ADDRESS="B" MIN="-90" MAX="110" PRIORITY="AUTO" HOME="0" VALUE="0" />
    <simple_rotary X="70.0" Y="0.0" Z="352.0" I="0" J="1" K="0" />
  </axis>
  <model_list>
    <dmf_file>
      <path FILE="elemek/B.dmf" />
      <rgb R="235" G="75" B="25" />
    </dmf_file>
  </model_list>

  <machine_part NAME="RobotAxis3">
    <axis>
      <control_info ADDRESS="C" MIN="-230" MAX="50" PRIORITY="AUTO" HOME="0" VALUE="0"/>
      <simple_rotary X="70.0" Y="0.0" Z="712.0" I="0" J="1" K="0" />
    </axis>
    <model_list>
      <dmf_file>
        <path FILE="elemek/C.dmf" />
        <rgb R="235" G="75" B="25" />
      </dmf_file>
    </model_list>

  <machine_part NAME="RobotAxis4">

```

30. ábra: MTD fájl adatstruktúra

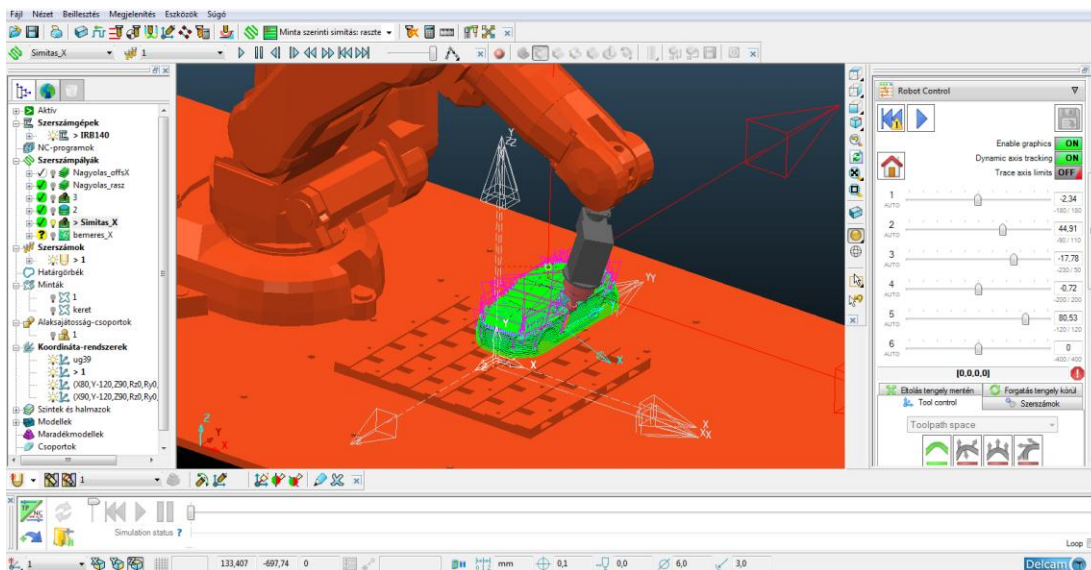
Az mtd fájl a következőket tartalmazza:

- A cellában szereplő összes elem CAD modelljének elérési útvonalát (a cellában szereplő eszközök, a robot elemei, illetve a főorsó).
- A kinematikai lánc felépítése, az egyes tengelyek helyzetének, és a hozzátartozó véghelyzetek megadásával
- Szerszám beillesztési pontja
- Munkadarab koordináta rendszere: alaphelyzetben a szoftver ide illeszti a munkadarabunkat, illetve ehhez képest módosíthatjuk elhelyezkedését
- Ütközési csoportokat: a cellában található egyéb eszközök, az ütközés vizsgálat ezekre is kiterjed

Nem adható meg azonban a soros kinematikától eltérő felépítésű lánc. Esetünkben ilyen volt az IRB1400h típusú robot, ezért a PowerMill-es szerszámpályákat az IRB 140-en teszteltük.

3), 4) Szimuláció és munkadarab elhelyezés:

Szerszámpálya generálást követően a teljes megmunkálást, beleértve a robotmozgást és az anyagleválasztást is, szimulálni tudjuk. Szintén szimuláció során ellenőrizhető a pályapontok elérhetősége is. Ehhez segítséget nyújt az egyes robotcsukló szélsőértékeinek feljegyzése.



31. ábra: PowerMill szimuláció

Mivel a Powermill lényegében egy CAM szoftver, lehetőség van teljes megmunkálás szimulációjára. Ez lehetővé teszi nagyolás során is az ütközés ellenőrzést, illetve annak automatikus elkerülését a szerszám bedöntésével.

PowerMill esetében is van lehetőség köztes „via” pont beszúrására az egyes szerszám pályák közé az a biztonságos pályaváltások érdekében.

4.3.Rapid Program elkészítése

Az ABB robotok a RAPID elnevezésű programnyelvet használják. Az ellenőrzött szerszám pályákat posztprocesszálni kell a robot számára. Process Simulate esetében ehhez rendelkezésre áll ún. OLP modul minden nagyobb robotgyártó termékeihez (ABB, KUKA, Yaskawa-Motoman, Fanuc, Stäubli, stb.) Szükséges továbbá koordináta rendszert megadni, amelyben a robotprogram íródik, ill. zónaadatokat kell beállítani.

4.3.1. Koordináta rendszer beállítása

Ez lehet a robot base koordináta rendszere, vagy munkadarab koordináta rendszere is. Előbbi esetében az előgyártmány helyzetét a szimulációnak megfelelően kell beállítani, utóbbinál a munkadarab bemérése szükséges. Ha a szimuláció során a robot egyes tengelyei többször is véghezlyezethez közel kerültek, a munkadarab nem a szimulációval megegyező pozícióban könnyen olyan helyzeteket eredményezhet, amelyek nem elérhetőek el a robot számára.

4.3.2. Zónaadatok megadása

Zónaadatok megadásával állítható be, hogy a robot mennyire közelítse meg az adott pozíciót. ABB esetében „fine”, vagy „z[érték]” parancsokkal adható meg. „Fine” esetében a robot pontosan felveszi az adott pozíciót, így meg is áll egy-egy pillanatra, vagyis szaggatott mozgást eredményez. Ez egyrészt megmunkálás szempontjából is kedvezőtlen, másrészt a megmunkálás idejét is növeli. Nagyobb „z” érték megadása esetében pontosság csökken, nem kívánt lekerekítések jöhetnek létre a gyártmányban. Legkedvezőbb „z”-t 0-ra állítani, ebben az esetben a robot nem fog megállni a pontban, egyenletes sebességgel halad tovább azon keresztül.

4.3.3. Utólagos módosítások

Nagyobb méretű (3-4 MB fölött) programok esetében előfordult, hogy a robotvezérlő memóriája megtelt, ezért nem tudta egyben megnyitni a létrehozott programot. Így azt kisebb részletekre kellett vágni.

Process Simulate esetében a létrehozott programok esetében az egyes pontok elnevezését le kellett rövidíteni, mert 16 bájtnál hosszabb elnevezéseket nem tud az S4C+ vezérlő kezelni.

4.4. Összesítés

4.4.1. Egyéb szempontok

Külső tengelyek kezelése:

Bonyolult geometriák, vagy nagy munkadarabok esetében szükség lehet külső tengely alkalmazására. Forgó asztal esetében több irányból válik hozzáférhetővé a munkadarab, így összetettebb geometriák is kimunkálhatók. Külső lineáris sín alkalmazásával jelentősen növelhető a robot munkatere, így az elkészíthető munkadarab mérete is.

➤ Process Simulate: A külsőtengelyeknek csak beállító, indexelt mozgását képes vezérelni, szimultán vezérlés nem valósítható meg

➤ PowerMill: 12 tengely kezelését képes szimultán módon kezelni, így a külső tengelyek is egyszerűen vezérlehetők.

4.4.2. Összevetés marási szempontok szerint

Összességében robotokkal való marási műveletek programozására a Delcam PowerMill szoftvere bizonyult alkalmasabbnak. Ennek főbb okai:

- egy szoftveren belül végezhető mind a szerszámpálya generálási, mind a robotprogramozási rész
- szerszám orientáció beállításának korlátlan lehetősége
- nagyméretű pályákat is könnyen kezel
- a megmunkálás szimulációval nagyolás során is végezhető ütközésvizsgálat , automatikus szerszámdöntés az ütközések elkerüléshez
- külső tengellyel könnyen bővíthető a cella, majd azt szimultán képes kezelni (akár 12 tengelyig)

A programok készítése során felmerült nehézségek:

- 1400h speciális kinematikája nem hozható létre a szoftverben, így azt kezelni csak korlátozott mértékben tudja
- munkadarab elhelyezésre kevesebb támogatást ad, így nehezebb megtalálni az ideális pozíciót

Process Simulate elsősorban nem marási műveletekre lett kifejlesztve, főbb akadályai ilyen célú használatának:

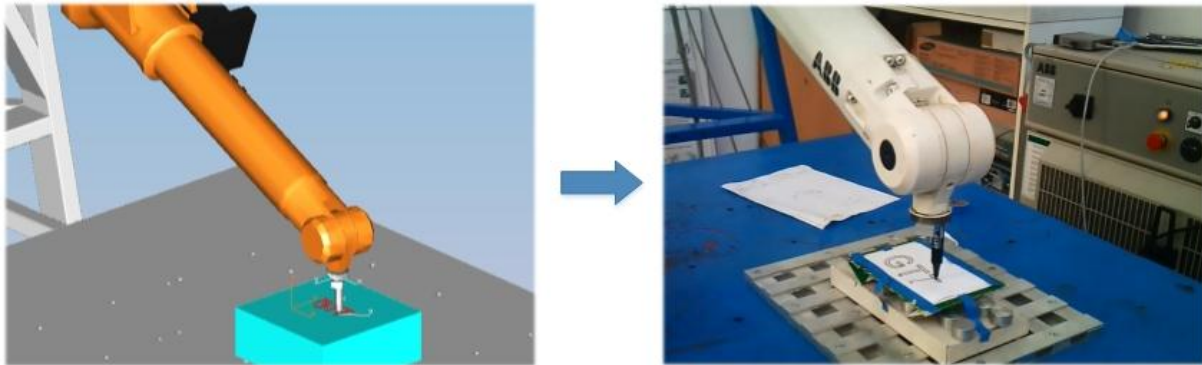
- Nagy (sok pontot tartalmazó) pályákat nehezen kezel
- Orientáció beállításra korlátozott lehetőségek vannak „CLS” fájlok betöltésénél
- Megmunkálás szimulációja külön szoftverben történik, elő gyártmánnyal való ütközés a robotos szimulációban nem ellenőrizhető
- Külső tengelyeket nem kezeli szimultán, csak beállító mozgások valósíthatók meg

Process Simulate alapvetően virtuális gyártás szimulációra lett kifejlesztve. Elsősorban hegesztési, ragasztási, vágási, gravírozási, és anyagmozgatási műveletekre használatos. Ezen műveletek pályái általában jóval kevesebb ponttal meghatározhatóak, így „CLS Upload” funkció itt használatos. Rendelkezik a korábban említett Elérhetőségi vizsgálat funkcióval, ezért könnyebb olyan pozíciót találni, ahol bejárható a teljes pálya.

5. Laboratóriumi kísérletek

Filctollas rajzolás

Első kísérletünk egy egyszerű „GTT” felirat elkészítése volt az IRB 1400h segítségével. A robotprogram elkészítésére Process Simulate-t használtuk.



32. ábra: Filctollas rajzolás

Ezzel az egyszerű rajz feladattal elsődleges célunk a teljes folyamat betanulása, letesztelése volt. Kezdvé a modell felirat létrehozásától, majd arra pálya illesztés NX CAM részében. CLS formátumként való átvitele Process Simulate-be. Szimuláció után a RAPID program feltöltése a robotra, illetve annak futtatása, és ellenőrzése.

Ezek a kísérletek mutatták meg, hogy „Fine” zónaadat megadása a robot szaggatott mozgását eredményezi, így marási kísérleteknél már ügyeltünk erre.

Szerszámbemérés

A pontos 5 tengelyes megmunkáláshoz elengedhetetlen a pontos szerszámbemérés. Esetünkben a rövid szerszámok nem ütköznek fel a főorsón belül, így kinyúlási hossz minden befogásnál változhat. Továbbá a főorsó forgási tengelye sem esik egybe a robot szerszámkoordináta rendszer Z tengelyével, ezért a pontos X és Y irányú szerszámeltolás értékeket is meg kellett határoznunk.

a) Bemérés az ABB 4 pontos módszerével

A robotvezérlő rendelkezik szerszámbemérő funkcióval. A robotmunkaterében egy kiválasztott fix pontot megközelítjük négy különböző orientációból a szerszám végponttal, így a vezérlő a redundáns helyzet meghatározásnak köszönhetően képes kiszámolni a szerszám végpont helyzetét. Megbecsüli továbbá a bemérés átlagos, illetve maximum hibáját. A tesztek során 10-20 [mm]-es maximális hibákat is tapasztaltunk.

Mivel az adapter tárcsa tervezésénél azt feltételeztük, hogy a 4 pontos módszer elegendően pontos szerszámbemérést fog eredményezni, ezért a minél egyszerűbb legyárthatóság érdekében a főorsó oldali illesztését nem terveztük bele.

Új adapter tárcsát tervezünk ahol meg lett oldva a főorsó illesztése, illetve szigorúbb tűréseket írtunk elő, amiknek a következőben ismertetett módszerekben lesz jelentősége.

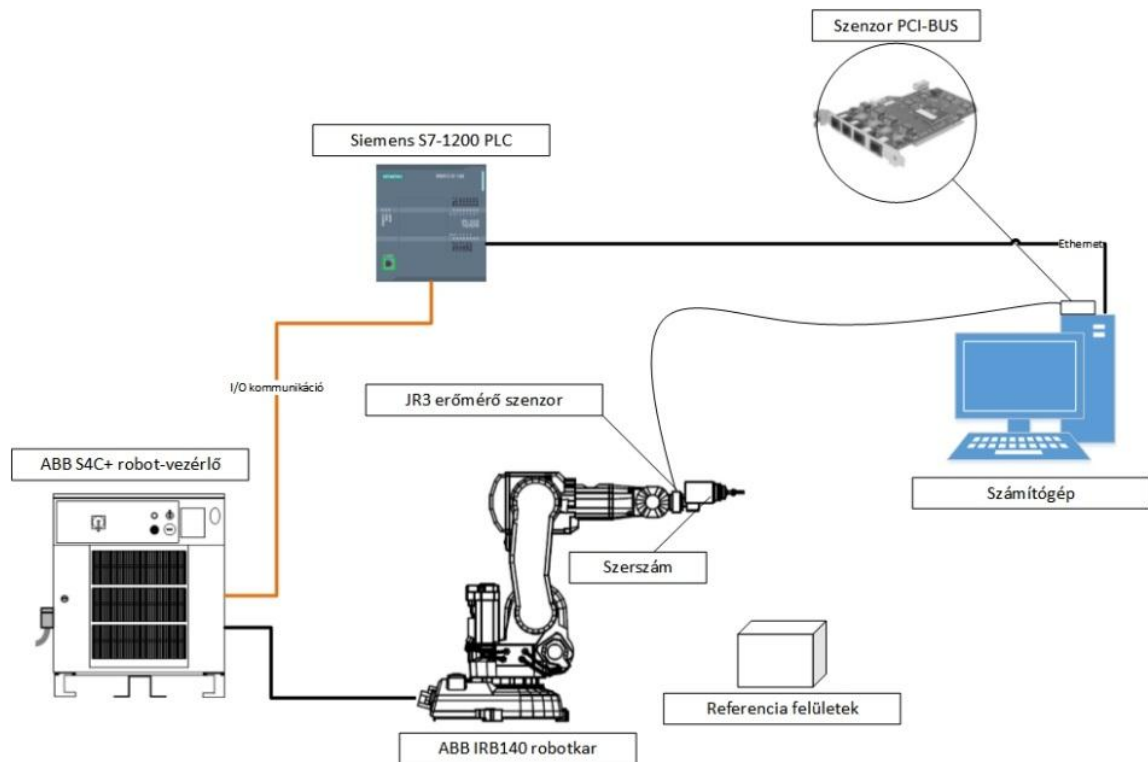
b) Bemérés koordináta mérőgép segítségével

A bemérés során az orsó és az adapter tárcsa összeszerelt állapotban volt. Ezzel megelőztük az összeszerelésből adódható pontatlanságokat. A szerszámkoordináta rendszer Z tengelyének meghatározására az adapter tárcsa palástfelület tette lehetővé. A marószerszám tengelyének bemérése patronba fordítva marószerszámmal (szerszámszár állt kifelé) lett elvégezve. Így ismert lett a két tengely egymáshoz képesti relatív helyzete. A robot szerszámkoordináta rendszerében történő abszolút méret meghatározáshoz a robotkarima felőli felfogató furatok is be lettek mérve. Mivel a robotkarimán 1 illesztő furat van, így a teljes helyzet-meghatározást nem biztosított. Így szükséges volt kidolgozni egy olyan módszert, amelynél a szerszám felszerelt állapotban bemérhető.

c) Bemérés JR3 erőmérő cella segítségével

Mivel az első 2 módszer egyike sem nyújtott kielégítően pontos eredményt, másik megoldást kellett találnunk. A szerszám beméréshez rendelkezésünkre állt a Tanszéktől kapott JR3 erőmérő szenzor. Továbbá szükségünk volt egy-egy referencia felületre, amelynek a helyzetét a robot base koordináta rendszerhez képest bemértük. A bemérés a robotra felszerelt adapter tárcsával a referencia felület megérintésével történt. Az érintkezési pontot a JR3 erőmérő szenzor jelének figyelésével vettük fel (lásd később). Így robot TCP aktuális helyzetének, és a referencia felületek helyzetének ismeretében ki tudjuk számolni a szerszám eltolást. X és Y irányokból szükség van a szerszám átmérőjének ismeretére is.

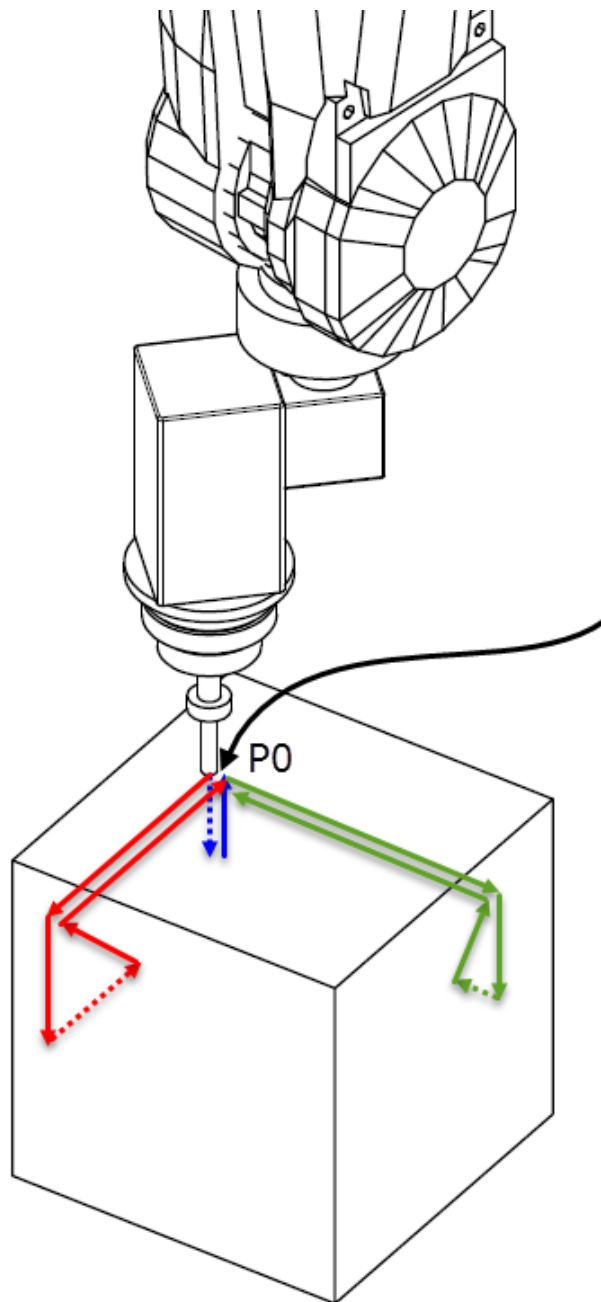
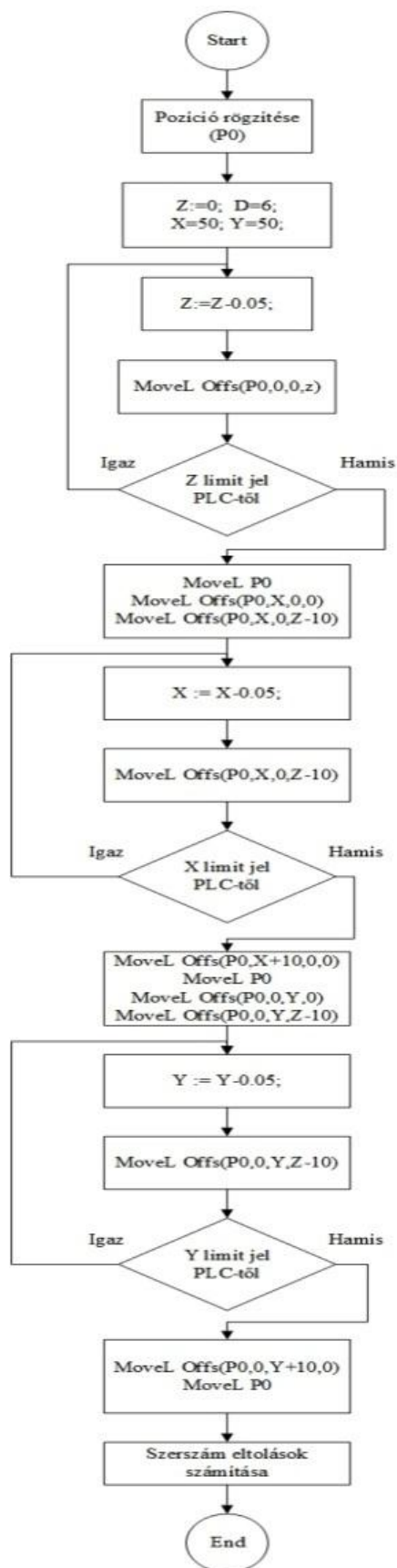
A szenzor jelét számítógép segítségével figyeltük, majd a robotot a lehető leglassabb sebességgel mozgatva megérintettük a felületet. Az érintkezési pontokban feljegyeztük a vezérlő által írt TCP koordináta értékeket. Ennek ismeretében már ki tudtuk számolni a szerszámkinyúlást. A fenti folyamatban azonban nagy az emberi bizonytalanság tényezője, így ezt célszerű lehet automatikusan elvégezni:



33. ábra: Automatikus szerszámbemérés architektúrája

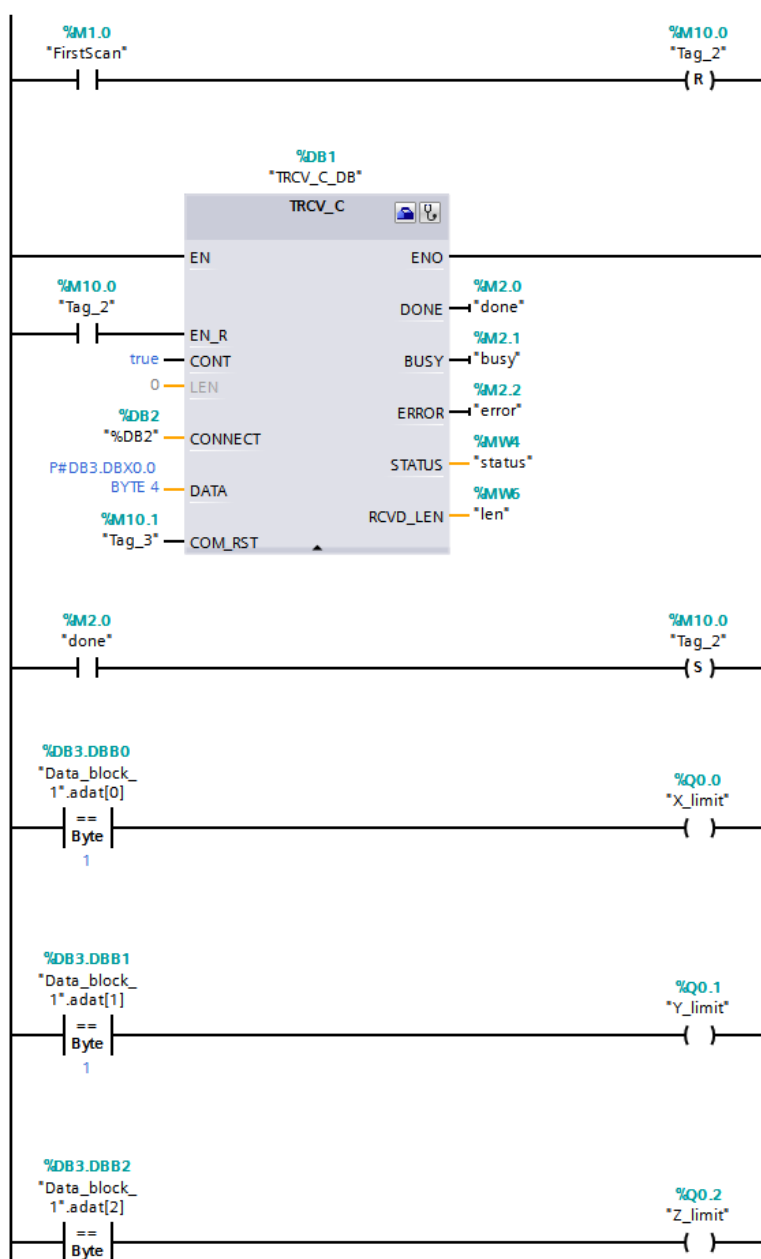
A bemérés elvégzésére írtunk egy RAPID programot, melynek a működését a 34. ábra szemlélteti.

A szerszámbeméréshez először szerszámot a referencia felület fölé mozgatjuk adott orientációban. Majd elindítjuk a RAPID programot, amely az ábra szerint megérinti az egyes referencia felületeket (34. ábra). Az érintkezési pont, illetve a referencia sík helyzetének ismeretében már képes kiszámolni a szerszám eltolást. Mivel program 0,05 [mm]-es léptékekben halad, így a szerszámbemérés pontossága is erre korlátozódik.



34. ábra: Szerszám bemérés folyamatábrája, mozgások illusztrációja

A működéséhez azonban szükség van a referencia felület érintését követően a vezérlésnek erről jelet küldeni. A jelet a robotkarima és a szerszám közé szerelt JR3 erőmérő szenzor szolgáltatja. A szenzor jelét Microsoft Visual Studio-ban írt szoftverben dolgozzuk fel, amely ha elér egy meghatározott limitet, jelzést küld Ethernet kapcsolaton keresztül a PLC-nek. A PLC a robot-vezérlővel digitális bemenetein keresztül kommunikál. A PLC programja TIA Portal V12 segítségével készült el, amelynek a létradiagramja a 35. ábrán látható.



35. ábra: PLC program létradiagramja

A szerszám eltolások számítása az alábbi módon történnek:

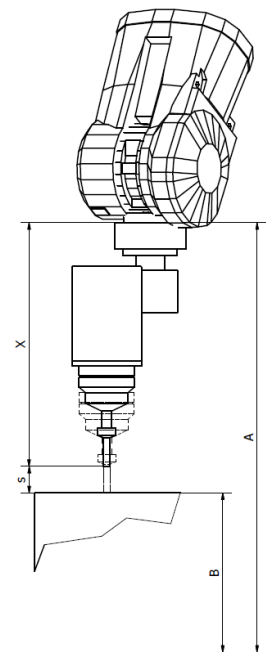
Jelmagyarázat:

- „A”: Robot TCP és Base koordináta rendszer távolsága főtengelyek mentén
- „B”: Referencia felület és Robot Base koordináta rendszer távolsága a főtengelyek mentén
- „S”: Szerszám és a referencia távolsága a kezdeti pontban
- „X”: Keresett távolság, a szerszám eltolás

Ezek alapján már egyszerűen számítható a szerszám kinyúlás:

$$X = A - B - S$$

Oldalirányú mérésnél figyelembe kell venni a szerszám átmérőjét is.



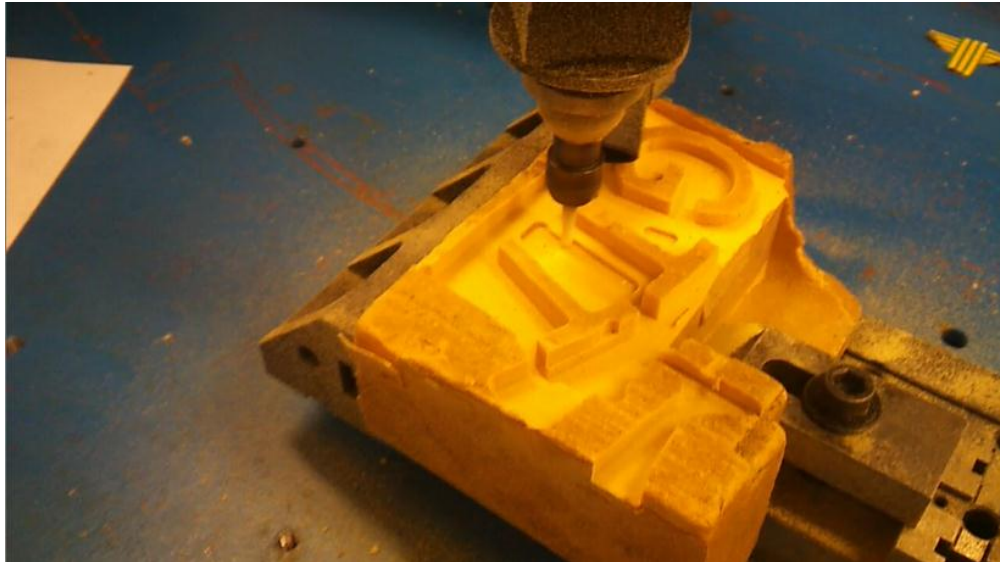
36. ábra: Szerszám bemérés

Munkadarab bemérés

Megmunkálások során mi a koordináta rendszernek a robot base koordináta rendszerét jelöltük ki, ami azt eredményezi, hogy a szimulációnak megfelelő pozícióba kell az elő gyártmányt helyezni. Ezért a megmunkálási pályákon felül minden modellhez készítettünk egy-egy további pályát, ami segítségünkre volt az elő gyártmány elhelyezésében. Ez a program az elő gyártmány élein futatta végig a szerszámot. Mivel a munkadarab elhelyezése így is hibával van terhelve, ezért előgyártmányra 4-5 mm-es további ráhagyást hagytunk.

GTT felirat

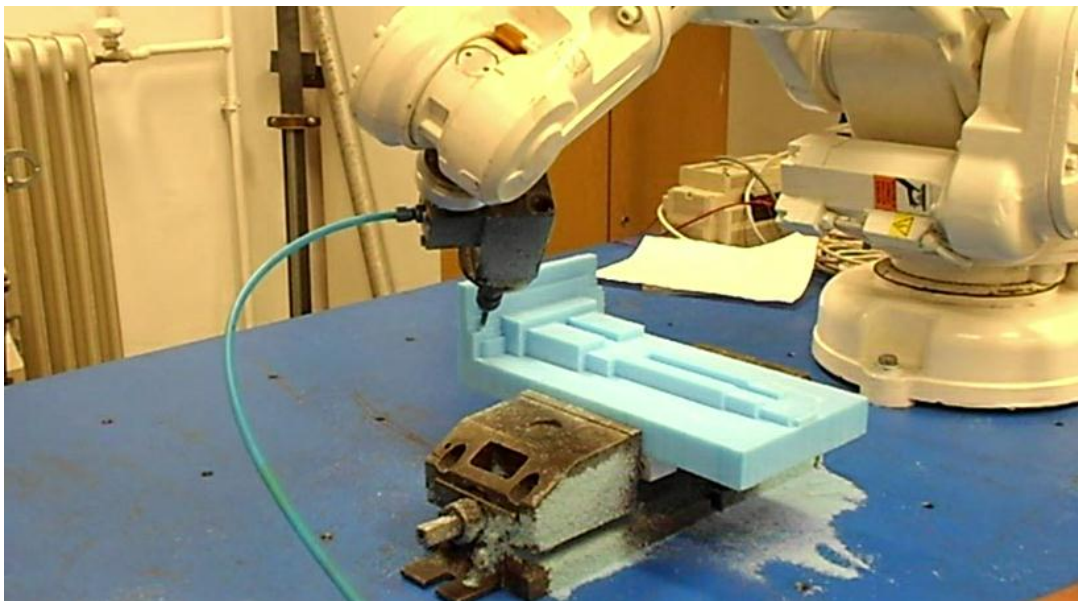
Az első marási kísérlet szintén egy „GTT” felirat kimunkálása volt poliuretán habból. A Process Simulate használata a nagymennyiségű pont miatt akadályba ütközött, így ennek a RAPID programját már PowerMill segítségével posztprocesszáltuk.



37. ábra: GTT marás

Empire State Building

Áramlástan Tanszék kérésére készült el az Empire State Building makettje szélcsatornás kísérletekhez, amely egyébként a Kutatók Éjszakáján is be lett mutatva az Áramlástan Tanszék laborjában. Az előgyártmány EPS szigetelőanyag volt.



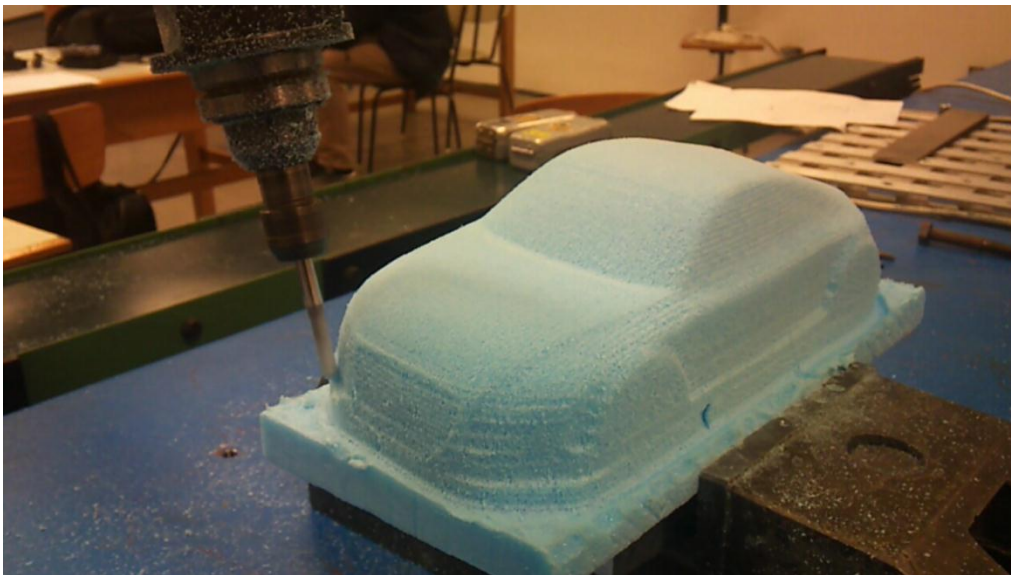
38. ábra: Empire State Building

A modell magassága miatt a robot nem fért volna hozzá kellően a hátsó oldalhoz, ezért 2 részre lett bontva, majd külön-külön történt a kimunkálás. Alapvetően a geometria 2,5D-s megmunkálás igényét veti fel, viszont a rendelkezésünkre állt szerszámok rövidsége miatt, egyes sarkok nem voltak hozzáférhetőek a függőleges szerszámhelyzetében. Ezeken a helyeken csak a szerszám döntésével tudtuk a maradékanyagot lemunkálni. Tehát ennél az egyszerű geometria kimunkálásánál is már 5 tengelyes marást kellett megvalósítanunk a munkadarab méretei miatt.

A nagyolási RAPID programot a robotvezérlő annak mérete miatt már nem tudta egyben megnyitni, így azt két részre kellett bontani, majd külön-külön futtatni.

A megmunkálásról készítettünk videót, amely a mellékelt DVD-n megtalálható.

Audi TT modell

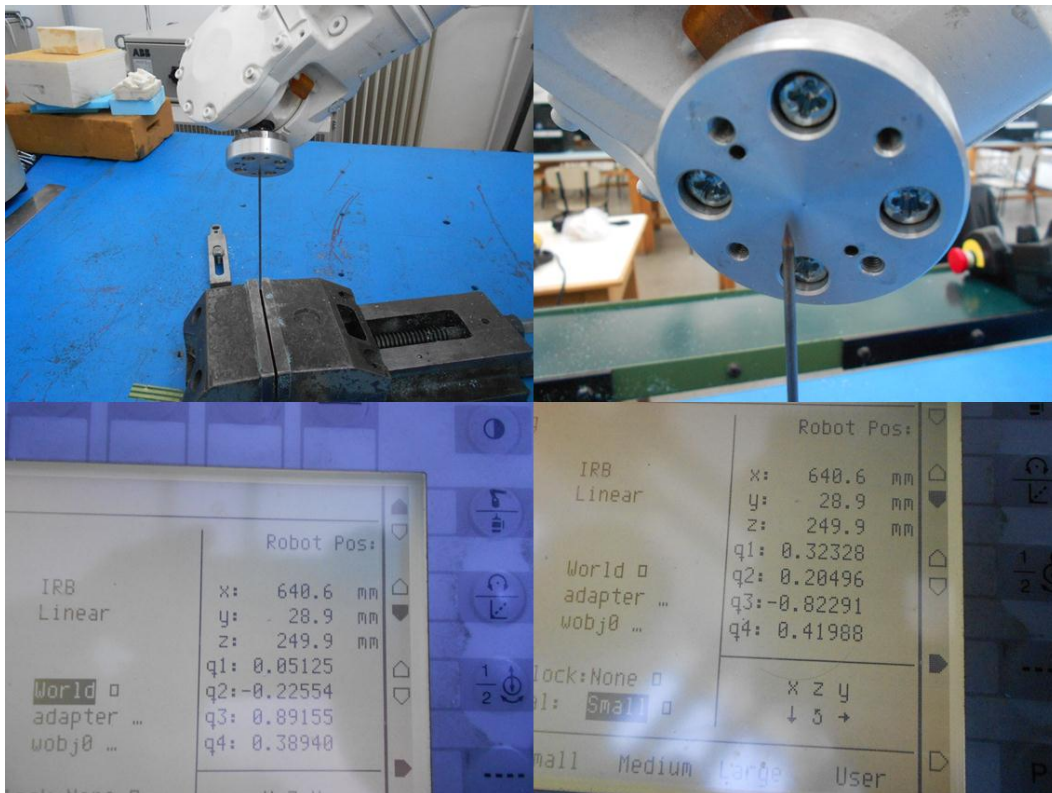


39. ábra: Audi TT marás

Szintén szélcsatornás kísérletekhez készült az Áramlástan Tanszék számára EPS szigetelőanyagból. A modell 1:15 arányban kicsinyített. A szerszám rövid kinyúlása miatt itt is szükségünk volt 5 tengelyes stratégiára. A megmunkálásról készítettünk videót, amely a mellékelt DVD-n megtalálható.

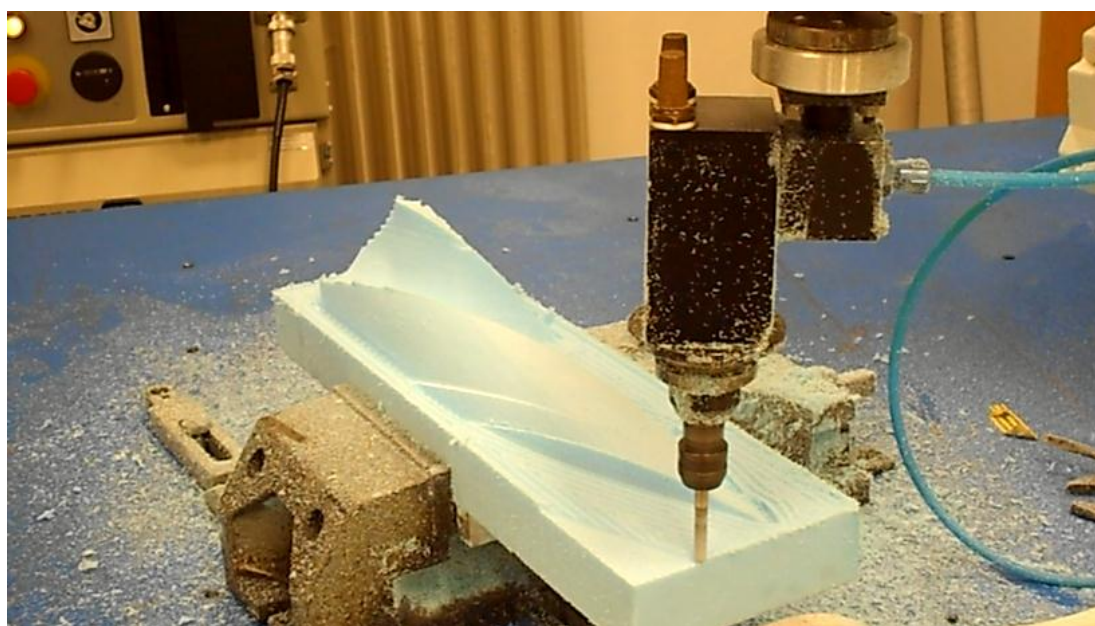
A marás során modell körbejárásához a robotnak konfigurációt kellett váltania, ami a marás pontosságára is hatással volt. Ahogy szakirodalmi áttekintésben is említettük, a robot különböző konfigurációiban eltérő az abszolút pontosság. Esetünkben, simításnál a modell egyik oldalán a szerszám programozott pontja a valódi pályapontokat vette fel, míg a másik oldalon a szerszám nem érintkezett a megmunkálandó felülettel. Feltehetően a konfigurációváltás okozta ezt a jelenséget. Ennek alátámasztására elvégeztünk egy egyszerű kísérletet.

A robot szerszámkarimájának középpontjával (robot TCP) felvettük egy tűske csúcspontját, majd megváltoztattuk az orientációt. A 40. ábrán jól látható, hogy több milliméteres eltérések is adódhatnak különböző orientációkban, miközben a Teach Pendant azonos koordináta értékeket mér.



40. ábra: Konfiguráció váltás hatása az abszolút pontosságra

Csigafúró



41. ábra: Csigafúró tanszéki tábla

Készítettünk egy dísz tárgyat Gyártástudomány- és technológia Tanszék számára. A kimunkálendő modell egy megdöntött csigafúró volt, amelynek palástfelületére a Tanszék nevét kellett horonymarással kiírni. Az előgyártmány anyaga itt is EPS szigetelőanyag volt. A megmunkálást az eddigi marásoknál is használt $\varnothing 6$ mm-es gömbmaróval végeztük. A felirat elkészítéséhez egy $\varnothing 2$ mm-es horonymarót használtunk. A feladat során a teljes maradékanyag eltávolításához több helyen is alámarást is kellett alkalmazni. Viszont ehhez, a szerszámot helyenként a vízszinttől 20-30°-os pozícióba kellett dönteni, amiből következik, hogy ezekben a helyzetekben a robot konfigurációja is jelentősen változott. A konfigurációváltások abszolút pontosságra gyakorolt negatív hatása az elkészült munkadarabon is látszódnak. (42. ábra).

A megmunkálásról készítettünk videót, amely a mellékelt DVD-n megtalálható.



42. ábra: Az orientáció váltásokból származó pontatlanság a munkadarabon

6. Robotos marás alkalmazási lehetőségek

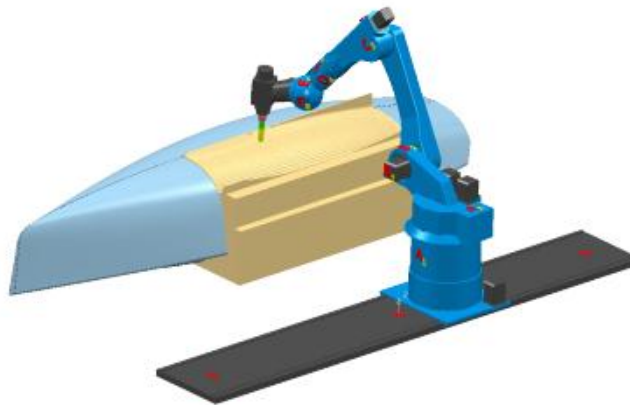
A dolgozatunkban ismertetett technológia olyan területeken játszhat fontos szerepet, amelyeket kis darabszám, egyedi, bonyolult szobor felületek, nagyméretű alkatrészek jellemeznek. A korábban említett robot tulajdonságok miatt (pl.: robot merevsége), elsősorban könnyen megmunkálható anyagok forgácsolására alkalmazható pl.: habok, műfa, gipsz.

Néhány alkalmazási példa:

- Szobrászat
- Filmipar díszleteinek elkészítése
- Mesterminta elkészítése különböző formaadó szerszámokhoz: homokforma, szilikon
- Méretarányos makettek készítése
- Kompozit termékek készítéséhez szükséges forma
- Szendvics szerkezetekhez maganyagának kialakítása



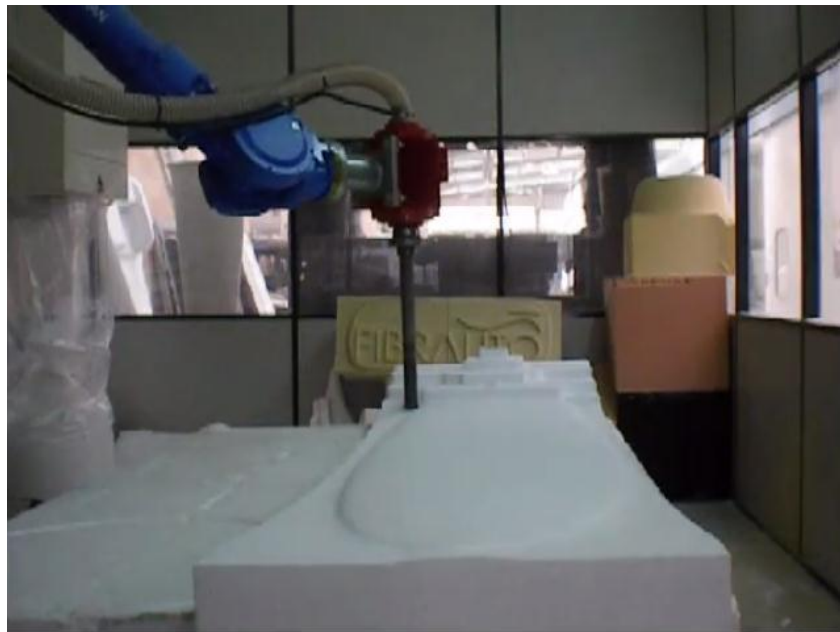
43. ábra: Makett készítése[33]



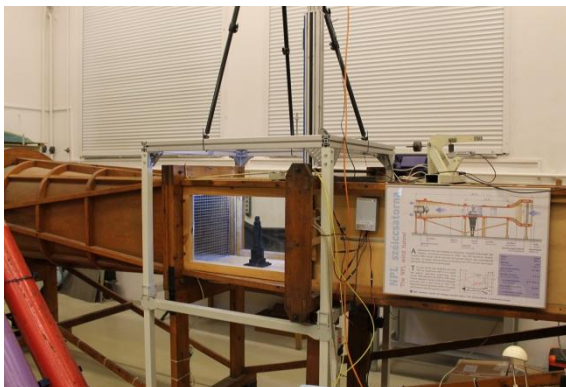
44. ábra: Hajótesthez forma gyártása [32]



45. ábra: Szobor felületek készítése [34] **46. ábra:** Szobor felületek készítése [35]



47. ábra: Szörfdeszka szendvics szerkezet belső maganyagának kimunkálása [36]



48. ábra: Modell szélcsatornás kísérletekhez

7. Fejlesztési lehetőségek

- További komolyabb munkák elvégzésének elsődleges feltétele a robot konfigurációváltásaiból adódó pontatlanság mértékének pontos meghatározása és kompenzálása. Egy részleges megoldás lehet a munkatér kibővítése külső tengellyel, ezzel megnövelve az optimális konfigurációban megmunkálható munkadarab méretet. A külső tengely alkalmazása nem csak a nagyobb méretű, de bonyolultabb alakzatok marását is lehetővé tenné.
- Gravírozás felületre: Pontos kalibrálása után robotot sokkal összetettebb feladatokhoz is lehetne alkalmazni. Ilyen lehet például, ha a szerszámbemérésnél ismertetett, JR3 erőmérő cellával történő automatikus érintkezés érzékelést egy koordináta mérőszoftverrel kombináljuk. Ez lehetővé tenné, hogy egy tetszőleges munkadarabot a robot munkaterében elhelyezve digitalizáljunk, majd a modellre gravírozandó feliratot vetíthessünk, amit szintén a robottal munkálnánk ki.
- Robot munkatér elkerítése a környezettől, illetve porelszívás automatizálása

8. Összefoglaló

TDK dolgozatunk célja ipari robotok mozgásprogramjainak létrehozása volt marási műveletekre, illetve ennek szoftveres lehetőségeinek megismerése. A kísérleteket a Gyártástudomány és –technológia Tanszék robotlaborjában található ABB IRB140, és IRB1400h típusú robotokon végeztük.

Dolgozatunkban 2 különböző offline robotprogramozási módszert vizsgáltuk meg közelebbről. Első esetben egy különálló CAM szoftverrel (Siemens NX 8.5) készítettük el a szerszámpályákat, majd egy off-line robotprogramozó, és szimulációs szoftver (Siemens Tecnomatix RobotExpert/Process Simulate) segítségével posztprocesszáltuk a robot RAPID programját. A másik esetben a Delcam PowerMill CAM rendszerét használtuk, mely rendelkezik Robot Interface kiegészítéssel, ami kifejezetten ipari robotok programozását segíti megmunkálási feladatoknál.

Első kísérleteink filctollas rajzolások voltak, ezzel egyrészt a mozgáspályák létrehozásának folyamatát, másrészt a robot kezelését tanultuk meg. A mozgásprogramokat NX 8.5, és PS segítségével hoztuk létre.

A tényleges megmunkálási pályákat már PowerMill szoftverben készítettük el az előző megoldásnál említett problémák miatt (nagy pályák nehézkes kezelése, orientáció beállítására korlátozott lehetőség). Az előgyártmány a robot kis merevégéből kifolyólag könnyen megmunkálható anyagból állt (EPS, illetve PUR hab).

A megmunkálásokhoz szükséges volt a szerszám kinyúlásának meghatározása. Ezt több módszerrel is elvégeztük. Végül az erőmérő szenzorral automatikusan is elvégezhetjük a mérést.

Szerszámpályák elkészítése során 2 dologra kellett különösen ügyelni, egyrészt orientációváltások nagy sebességeket eredményezhetnek a robot egyes csuklóiban, másrészt a különböző konfigurációban a robot abszolút pontossága jelentős mértékben eltérhet. Ezért az orientáció, és konfiguráció váltásoknál a valós szerszámpályák a programozottól eltérő helyzetekben is előfordultak, így a munkatér csak korlátozott részében tudtunk kielégítő pontosságú megmunkálást végezni.

9. Források

- [1] J. Pandrenemenos, C. Doukas, P. Stavropoulos, G. Chryssolouris.: Machining with robots: A critical review; 7th International Conference on Digital Enterprise Technology (2011)
- [2] Chris Felix: Robots as Precision Machine Tools 2012.01.23. Production Machining
- [3] “COMET” Project Targets Industrial Robots for High End Machining The COMET EU Project Newsletter Issue 1 2011 0
- [4] MC Bonney, PJ Edwards, JA Gleave, JL Green, RJ Marshall, YF Yong: The Simulation of Industrial Robot Systems, International Journal of Manufacturing Science, Vol 12, pp. 273-281 (1984).
- [5] Yonghua Chen and Fenghua Dong: Robot machining: recent development and future research issues International Journal of Advanced Manufacturing Technology, pp 1489-1797 (2012).
- [6] Suh, Y. S., & Lee, K.: NC milling tool path generation for arbitrary pockets defined by sculpture surfaces. Computer-Aided Design, Vol. 22, pp 273- 284 (1990).7
- [7] Huang, Y., & Oliver, J. H.: Non-constant parameter NC tool path generation on sculptured surfaces. International Journal of Advanced Manufacturing Technology, pp 281-290 (1994).
- [8] Chen, H., & Hu, Y.: Implementation of a Robot System for Sculptured Surface Cutting. Part 1. Rough Machining. International Journal of Advanced Manufacturing Technology, pp 624-629 (1995).
- [9] Vergeest, J. S., & Tangelder, J. W.: Robotmachines rapid prototype. Industrial Robot, pp 17-20 (1996).
- [10] Song, Y., & Chen, Y. H.: Feature-based robot machining for rapid prototyping. Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture 1999, pp 451 (1999).
- [11] S. Mitsi, K.-D. Bouzakis, G. Mansour, D. Sagris, G. Maliaris: Off-line programming of an industrial robot for manufacturing, Int J Adv Manuf Technol (2005) 26: 262–267
- [12] Mark Albert: Programming A Robot The Way You Program A CNC Machine Tool <http://www.mmsonline.com/articles/programming-a-robot-the-way-you-program-a-cnc-machine-tool> (2014.09.13.)
- [13] A.Y. Elatta, Li Pei Gen, Fan Liang Zhi, Yu Daoyuan and Luo Fei: An Overview of Robot Calibration Information Technology Journal 3(1): 74-78 (2004).
- [14] Driels MR, Swayze W, Potter S.: Full-pose calibration of a robot manipulator using a coordinate measuring machine. Int J Manuf Technol Robot Comput Integr Manuf 27(4):881–888 (2011) 18:34–41 (1993)

- [15] Oki, Y., & Kanitani, K.: Development of robot machining system for aluminum building materials. *J. JSME* 99, pp 78 (1996)
- [16] Matsuokaa, S.-i., Shimizua, K., Yamazakib, N., & Okib, Y.: High-speed end milling of an articulated robot and its characteristics. *Journal of Materials Processing Technology*, Vol. 95, pp 83-89 (1999).
- [17] Young, K., & Pickin, C. G.: Accuracy assessment of the modern industrial robot. *Journal of Industrial Robot*, Vol. 27, pp. 427-436 (2000).
- [18] Kieffer, J., Cahill, A., and James, M.: Robust and accurate time-optimal path tracking control for robot manipulator, *IEEE Trans. Robotics Automat.* 6, 880–890 (1997).
- [19] Hrman, M., Kunica, Z., and Vranjes, B.: Positions of parts with vertical assembly paths at automatic assembly system planning, in: *Intelligent Manufacturing and Automation: Proc. of Internat. Symp. DAAAM International*, Vienna, Austria, pp. 199–200 (1999).
- [20] Veryha, Y., & Kurek, J.: Application of Joint Error Mutual Compensation for Robot End-effector Pose Accuracy Improvement. *Journal of Intelligent and Robotic Systems*, Vol. 36, pp 315–329 (2003).
- [21] Hui Z., Jianjun W., George Z., Zhongxue G., Zengxi P., Hongliang C., Zhenqui Z: Machining with Flexible Manipulator: Toward Improving Robotic Machining Performance, *IEEE/ASME International Conference on Advanced Intelligent Mechatronics* (2005).
- [22] Vosniakos G, Matsas E.: Improving feasibility of robotic milling through robot placement optimization. *Robot Comput Integr Manuf* 26:517–525 (2010).
- [23] Andres J, Gracia L, Tornero J.: Calibration and control of a redundant robotic workcell for milling tasks. *Computer-Integrated Manufacturing* 24(6):561–573 (2011).
- [24] C. Doukas, J. Pandremenos, P. Stavropoulos, P. Fotinopoulos, G.Chryssolouris.: On an empirical investigation of the structural behaviour of robots http://cometproject.eu/downloads/news/2012-06-CIRP_CMS_45_Empirical_investigation_of_the_structural_behaviour_of_robots_June_2012.pdf (2014. 08. 24)
- [25] Christian L., Marcel H., Dirk E., Daniel O.: Milling with industrial robots: Strategies to reduce and compensate process force induced influences Mechanism http://cometproject.eu/publications/etfa2012_a34.pdf (2014. 08. 24)
- [26] Olof S., Björn O., Ulrich S., Anders R., Arnold P., Rolf J.: High-Accuracy Milling with Industrial Robots using a Piezo-Actuated High-Dynamic Compensation Mechanism <http://cometproject.eu/publications/icra2012.pdf> (2014. 08. 24)
- [27] <http://new.abb.com/products/robotics/industrial-robots/irb-140> (2014.09.29)
- [28] Product On-line Manual IRB 1400 3HAC 2914-1 M98 (2014.08.24)
- [29] www.jr3.com (2014.08.24)
- [30] <http://www.plm.automation.siemens.com/> (2014.08.24)

- [31] <http://www.cnc.hu/2013/05/robot-programozas-powermill-kornyezetben/> (2014.08.12)
- [32] www.delcam.com (2014.08.24)
- [33] <http://www.robotsinarchitecture.org/> (2014.10.20)
- [34] <http://www.sprutcam.com/> (2014.08.24)
- [35] <http://www.mmsonline.com/blog/post/machining-with-robots> (2014.08.24)
- [36] <http://www.fibrauto.com/videos> (2014.10.20)

10. Melléklet

<https://drive.google.com/folderview?id=0BztBkfaJFq7YYIFTcmdHajRzYk0&usp=sharing>

DVD melléklet tartalma:

{1} Empire State Building marása

EmpireState.mp4

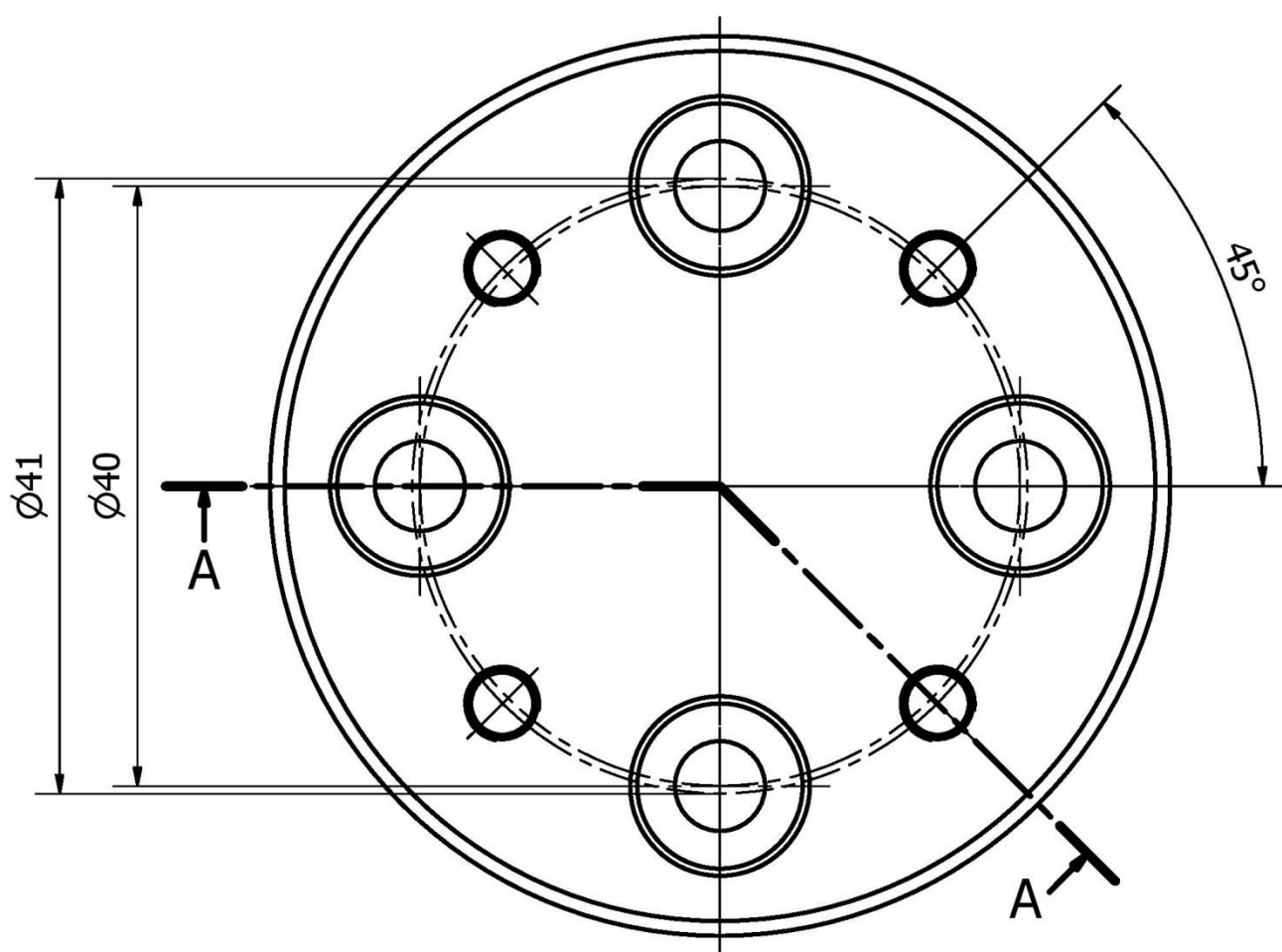
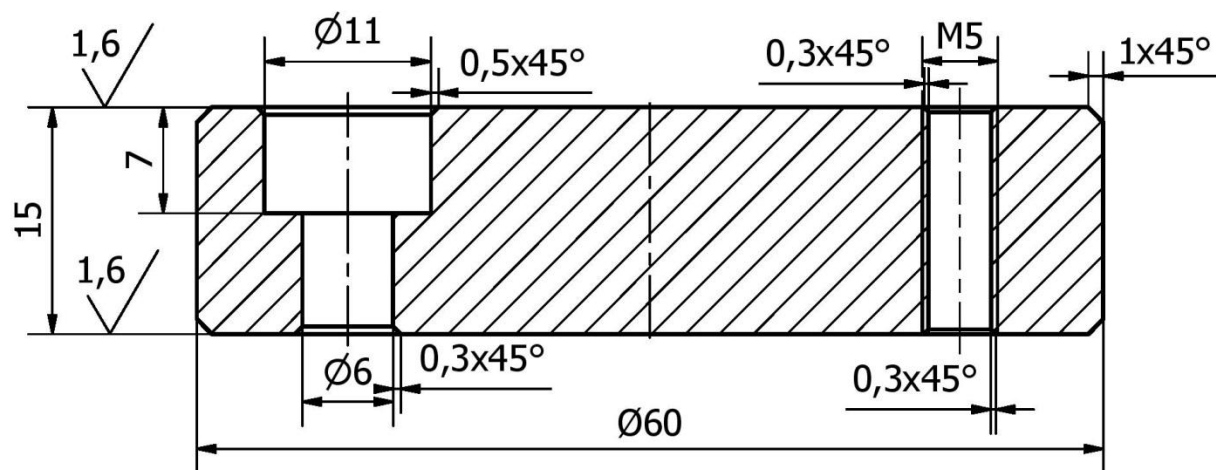
{2} Audi TT marása

AudiTT.mp4

{3} Csigafűró marása

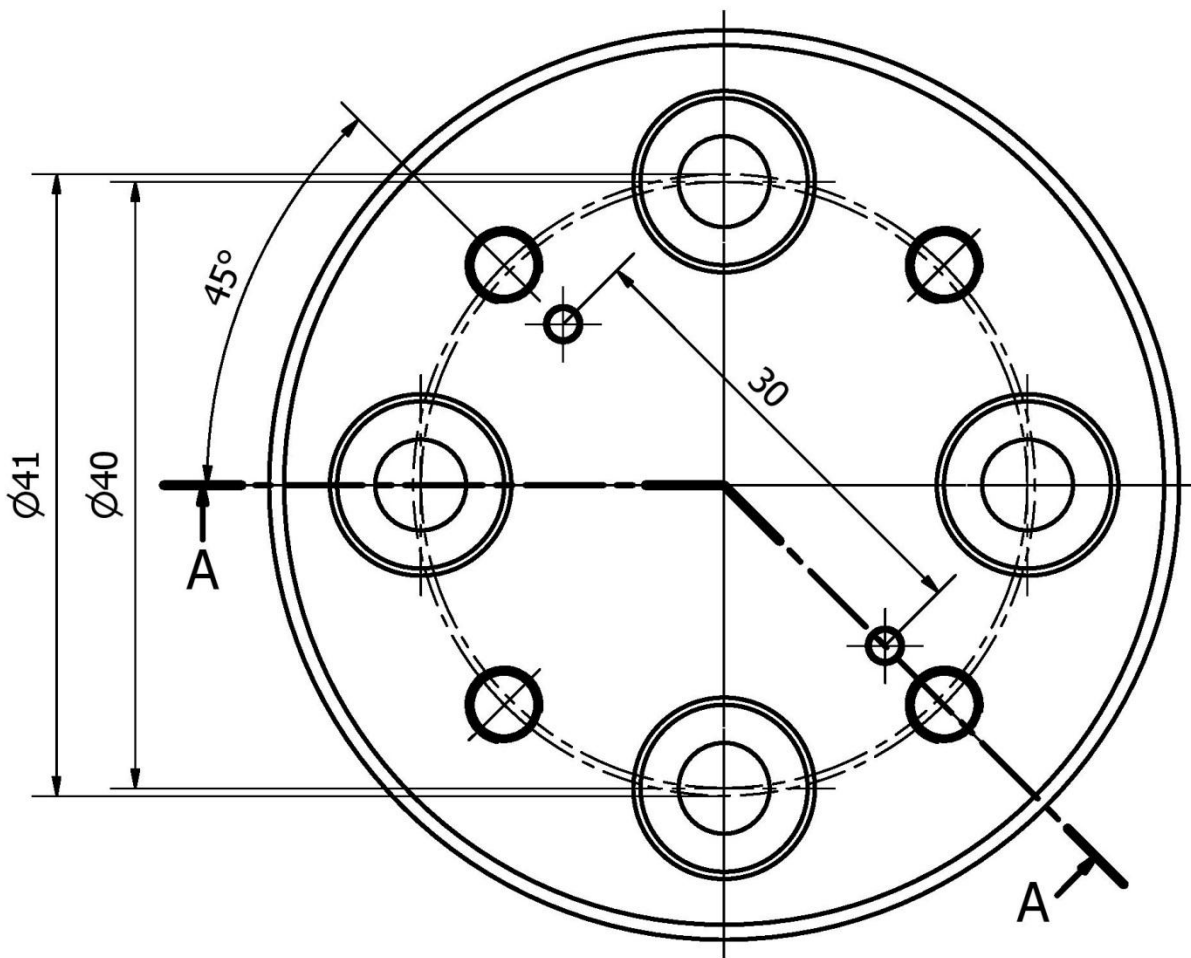
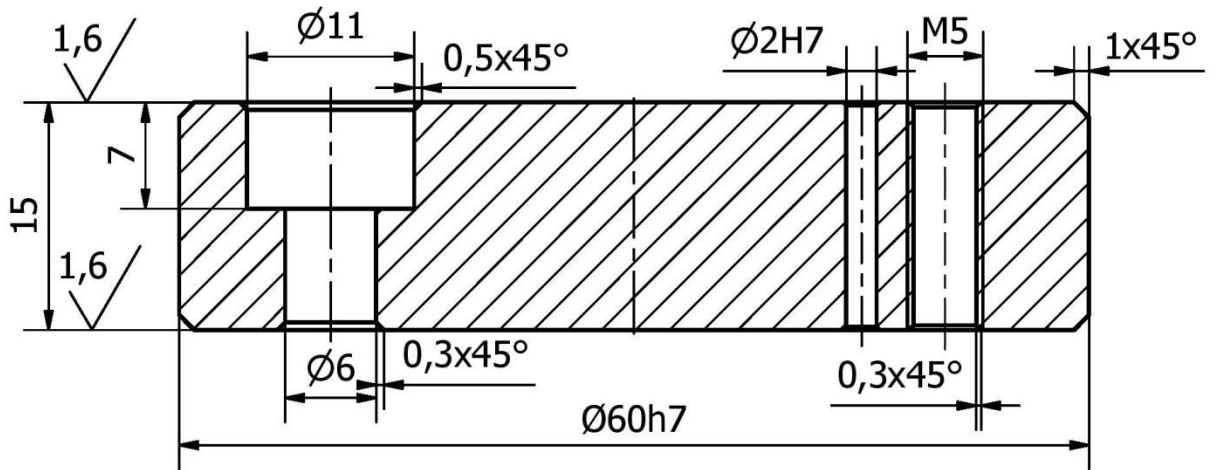
Csigafuro.mp4

A-A



Tervező: Rostási Gergő Nyemcsok Máté	Gyártmány: Robot megmunkáló		Méret- arány: 2:1	BME Gyártástudomány és -technológia Tanszék
Dátum: 2014.03.14.	Megnevezés: Adapter tárcsa		Vet. mód:	
Ellenőrizte: Tóth András	Anyag: Aluminium	Tömeg: 0,103 kg		

A-A



Tervező: Rostási Gergő Nyemcsok Máté	Gyártmány: Robot megmunkáló		Méret- arány: 2:1	BME Gyártástudomány és -technológia Tanszék
Dátum: 2014.05.22.	Megnevezés: Adapter tárcsa			
Ellenőrizte: Tóth András	Anyag: Aluminium	Tömeg: 0,103 kg	Vet. mód: 	