



PANNON EGYETEM

VEGYÉSZMÉRNÖKI- ÉS ANYAGTUDOMÁNYOK DOKTORI ISKOLA

**Ipari mobil robotok kiválasztási módszerének szakértői rendszer
alapú támogatása**

DOI:10.18136/PE.2025.973

DOKTORI (PhD) ÉRTEKEZÉS

KÉSZÍTETTE:

Boleraczki Miklós

okleveles gépészmérnök

okleveles mérnök-informatikus

TÉMAVEZETŐ:

Dr. Gyurika István Gábor

egyetemi docens

Pannon Egyetem

Mérnöki Kar

Műszaki Tudományok Kutató-Fejlesztő Központ

2025.

Ipari mobil robotok kiválasztási módszerének szakértői rendszer alapú támogatása

Az értekezés doktori (PhD) fokozat elnyerése érdekében készült a Pannon Egyetem Vegyészmérnöki- és Anyagtudományok Doktori Iskolája keretében

Anyagtudományok és Technológiák tudományágban

Írta: Boleraczki Miklós

Témavezető: Dr. Gyurika István Gábor

Elfogadásra javaslom: igen / nem.

.....
témavezető

Az értekezés bírálatra bocsátható.

.....
TDHT
elnök

A jelölt az értekezés nyilvános vitáján %-ot ért el.

A bíráló Bizottság tagjai:

elnök:.....

bírálok:.....

tagok:.....

Veszprém,

.....
Bíráló Bizottság elnök

A doktori (PhD) oklevél minősítése:.....

Veszprém,

.....
EDHT
elnök

Rövidítések

AI – mesterséges intelligencia (Artificial Intelligence)
AMR – mobil robot (Automated Mobile Robot)
AGV – robotkocsi (Automated Guided Vehicle)
CNC – számítógépes számjegy vezérlés (Computer Numerical Control)
CPS – Kiber-fizikai rendszer (Cyber-Physical Systems)
DSS – döntéstámogató eszköz (Decision Support System)
FIFO – elsőre be, elsőre ki (First In, First Out)
FMS – rugalmas gyártórendszer (Flexible Manufacturing System)
DES – diszkrét eseményű rendszerek (Discrete Event Systems)
DFAM – additív gyártáshelyes tervezés (Design for Additive Manufacturing)
DS – digitális árnyék (Digital Shadow)
DT – digitális iker (Digital Twin)
ET – szakértői rendszer (Expert System)
IoS – szolgáltatások internete (Internet of Service)
IoT – dolgok internete (Internet of Things)
KBS – tudás alapú rendszer (Knowledge Based System)
MCDM – többváltozós kritérium alapú döntéshozatal (Multi Criteria Decision Making)
OEE – általános eszközkihasználtság (Overall Equipment Effectiveness)
OEEML – gyártósor általános eszközhatékonyság (Overall Equipment Effectiveness of a Manufacturing Line)
OLE – általános sor hatékonyság (Overall Line Effectiveness)
OPE – általános üzemhatékonyság (Overall Plant Effectiveness)
RMS – újra konfigurálható gyártórendszer (Reconfigurable Manufacturing System)
ROS – robot operációs rendszer (Robot Operating System)
ROI – befektetés megtérülése (Return on Investment)
RTLS – valós idejű lokációs rendszer (Real Time Location System)
SD – rendszer dinamika (System Dynamics)
SLAM – párhuzamos navigálás és térképkészítés (Simultaneous Localization and Mapping)

Fogalmak

big data – nagyméretű strukturálatlan adathalmaz
layout – elrendezési vázlat, gyártógépek egymáshoz viszonyított helyzete

Tartalomjegyzék

Rövidítések.....	4
Fogalmak.....	4
Kivonat.....	7
Abstract.....	8
Zusammenfassung.....	9
1 Irodalmi áttekintés.....	10
1.1 Ipar 4.0.....	10
1.2 A gyártórendszerek tervezése.....	13
1.3 A gyártórendszerek elemzése és szimulációja.....	14
1.4 Petri-hálók.....	15
1.5 Minősítés szimulációval.....	17
1.6 Gyártórendszerek optimalizálása.....	19
1.7 Gyártórendszer minősítéséhez mérőszámok.....	21
1.8 A gyártás minősítéséhez mérőszámok.....	23
1.9 A gyártási környezet várható jövőbeli trendjei.....	26
1.10 Mobil robotok.....	27
1.11 Ipari mobil robotok.....	29
1.12 Kihívások a robotika különböző területein.....	32
1.13 Mesterséges intelligencia, szakértői rendszerek.....	33
1.14 Multi kritériumos kiválasztási módszerek.....	40
2 A Mobil robot központú modell fejlesztése.....	53
2.1 Anyagáramlás megvalósítása szállítószalaggal.....	53
2.2 Flexibilis gyártórendszer.....	54
2.3 Automatikus vezérlésű járművel kiegészített gyártórendszer.....	57
2.4 Robottal kiegészített modell.....	58
2.5 Gyártórendszer és mobil robot modellek kapcsolódási pontjai.....	63
2.6 Esettanulmányok a gyártórendszer modellhez.....	69
2.7 Két pont közötti szállítás.....	70
2.8 Többféle termék gyártására esettanulmány.....	75
2.9 Kiértékelő függvényként való felhasználás.....	78
2.10 A módszer előnyei az esettanulmányok alapján.....	81
2.11 Újdonság tartalom és hozzájárulás.....	81
3 A szakértői rendszer felépítése.....	83
3.1 Az optimalizációs feladat megfogalmazása.....	83

3.2	Tervezési paraméterek a mobil robotok integrálásához.....	84
3.3	Különbségek a robotok között – adatbázis	85
3.4	Különbségek a robotok között – tudásbázis.....	86
3.5	Szakértői rendszer első esettanulmánya	89
3.6	Szakértői rendszer második esettanulmánya	91
3.7	Szakértői rendszer harmadik esettanulmánya	92
3.8	Újdonságtartalom és hozzájárulás	101
4	Jövőbeni tervek.....	103
	Összefoglalás.....	104
	Új tudományos eredmények: a doktori értekezése tézisei	105
	New scientific results: thesis statement of the doctoral dissertation.....	107
	Ábrajegyzék	109
	Táblázatok jegyzéke	110
	Irodalomjegyzék	111
	Mellékletek	118
	Köszönetnyilvánítás	120

Kivonat

A gyártórendszerekben az áruk előállítása megköveteli a hatékonyság növelését, ugyanakkor az igények az egyedi termékek irányába mozdulnak el, ezzel kétféle egymásnak ellentmondó követelményt jelentenek a gyártórendszerek fejlesztőinek és üzemeltetőinek. Az ipar 4.0 és 5.0 egyik központi eleme az ipari mobil robot, amely a termékek gyors és pontos gyártását ugyanakkor az egyedi termékek létrehozását támogató gyártástechnológiai eszköz. Dolgozatommal ilyen mobil robotok gyártórendszerbe integrálásának első lépéséhez a kiválasztáshoz kívánok segítséget nyújtani egy fejlett döntéstámogató eszköz alapjainak lerakásával.

A mobil robotokat különféle feladatokra lehet használni, úgymint alapanyag szállítás, késztermék szállítás, részegységek szállítása, selejtek elszállítása vagy akár összeszerelési platform. Ennek megfelelően kell kiválasztani a piacon elérhető robotokból a feladatra ideálisnak tekinthető mobil robotot.

Ehhez sorra vettem a robotok tulajdonságait, majd a felírható modelleket egyszerűsítettem, és így vontam le következtetéseket a választandó mobil robotra vonatkozólag. A dolgozatban három fő területet érintek. Az elsőben felépítem a mobil robot központú gyártórendszer fogalmát, a korábbi gyártórendszer fogalmakra alapozva. A másodikban ennek a szemléletnek a felhasználásával esettanulmányokkal bizonyítom ennek a módszernek a hatékonyságát. A harmadikban egy olyan módszert ismertetek, amely egy szakértői rendszerhez képes tudásbázist létrehozni egy adatbázisból, és szintén egy esettanulmánnyal tesztelem és igazolom, hogy ily módon hatékony kiválasztási metódus valósítható meg.

A fenti három terület összefüggően a mobil robotok gyártórendszerbe integrálásának feladatát hivatott megkönnyíteni, és így a dolgozat célja is olyan szemlélet kialakítása, amellyel ezen eszközök optimálisan integrálhatóvá válnak a jelen kor követelményeit minden szempontból kielégítő gyártórendszerekben.

Abstract

In manufacturing systems, the production of goods requires increased efficiency, while at the same time demand is shifting towards customized products, presenting two conflicting requirements for developers and managers of manufacturing systems.

One of the central elements of Industry 4.0 and 5.0 is the industrial mobile robot, which is a manufacturing technology tool that supports the fast and accurate production of customized products. With my thesis, I aim to help with the first step of integrating such mobile robots into manufacturing systems by laying the foundations for an advanced decision support tool.

Mobile robots can be used for a variety of tasks, such as transporting raw materials, transporting finished products, transporting components, transporting rejects, or even as an assembly platform. Accordingly, the mobile robot that is considered ideal for the task must be selected from among the robots available on the market.

To do this, I listed the characteristics of the robots, then simplified the models that could be written down, and thus drew conclusions about the mobile robot to be selected. In this thesis, I cover three main areas. In the first, I build the concept of a mobile robot-centric manufacturing system based on previous manufacturing system concepts. In the second part, I demonstrate the effectiveness of this method using case studies. In the third part, I describe a method that can create a knowledge base for an expert system from a database, and I also test and verify with a case study that an effective selection method can be implemented in this way.

The above three areas are interrelated and are intended to facilitate the integration of mobile robots into manufacturing systems. Thus, the aim of this thesis is to develop an approach that allows these devices to be optimally integrated into manufacturing systems that meet all the requirements of the present day.

Zusammenfassung

In Fertigungssystemen erfordert die Herstellung von Waren eine Steigerung der Effizienz, während sich die Nachfrage gleichzeitig in Richtung individueller Produkte verschiebt, was für die Entwickler und Betreiber von Fertigungssystemen zwei widersprüchliche Anforderungen bedeutet. Ein zentraler Bestandteil von Industrie 4.0 und 5.0 ist der industrielle mobile Roboter, ein Fertigungstechnologie-Werkzeug, das die schnelle und präzise Herstellung von Produkten unterstützt und gleichzeitig die Herstellung individueller Produkte ermöglicht. Mit meiner Arbeit möchte ich einen Beitrag zur Auswahl solcher mobiler Roboter leisten, die als erster Schritt in die Integration in Fertigungssysteme dienen, indem ich die Grundlagen für ein fortschrittliches Entscheidungshilfetool schaffe.

Mobile Roboter können für verschiedene Aufgaben eingesetzt werden, wie z. B. den Transport von Rohstoffen, Fertigprodukten, Bauteilen, Ausschussware oder sogar als Montageplattform. Dementsprechend muss aus den auf dem Markt erhältlichen Robotern der für die jeweilige Aufgabe als ideal anzusehende mobile Roboter ausgewählt werden.

Zu diesem Zweck habe ich die Eigenschaften der Roboter aufgelistet, dann die in Frage kommenden Modelle vereinfacht und so Schlussfolgerungen hinsichtlich des auszuwählenden mobilen Roboters gezogen. In der Arbeit behandle ich drei Hauptbereiche. Im ersten Bereich entwickle ich das Konzept eines mobilen roboterzentrierten Fertigungssystems auf der Grundlage früherer Konzepte von Fertigungssystemen. Im zweiten Teil belege ich anhand von Fallstudien die Wirksamkeit dieser Methode unter Verwendung dieses Ansatzes. Im dritten Teil stelle ich eine Methode vor, mit der aus einer Datenbank eine Wissensbasis für ein Expertensystem erstellt werden kann, und teste und bestätige ebenfalls anhand einer Fallstudie, dass auf diese Weise eine effektive Auswahlmethode realisiert werden kann.

Die drei oben genannten Bereiche sollen zusammen die Integration mobiler Roboter in Fertigungssysteme erleichtern. Das Ziel der Arbeit ist es daher, einen Ansatz zu entwickeln, mit dem diese Geräte optimal in Fertigungssysteme integriert werden können, die den Anforderungen der heutigen Zeit in jeder Hinsicht gerecht werden.

1 Irodalmi áttekintés

Jelen disszertáció célját úgy fogalmazom meg, hogy egy korszerű döntéstámogató eszköz fejlesztésének elméleti megalapozása legyen kiegészítve gyakorlati esettanulmányokkal és tesztekkel. A kutatás tárgya az Ipar 4.0 gyártórendszerekben használatos ipari mobil robotok kiválasztásának támogatása, megadott robot adatbázisból tudásbázis létrehozása, majd a tudásbázis alapján a gyártórendszer követelményeinek leginkább megfelelő optimális mobil robotra javaslattevés.

1.1 Ipar 4.0

A növekvő termelékenység minden ipari forradalom magja [1].

A növekvő nemzetközi verseny, az egyedi termékek iránti kereslet és a termékek rövidebb életciklusa olyan kihívások elé állítják a vállalatokat, amelyeket hagyományos termelési rendszerrel már nem lehet teljesíteni [2]. Ez hívta életre az Ipar 4.0 koncepciót, amely hangsúlyt fektet az intelligens termékek és folyamatok létrehozására. Ebben a cél az egyedi termékek gyártása, a tömegtermelésre jellemző hatékonysággal. A hagyományos raktárra való gyártás helyett a vevői rendelésre történő termelés az új megközelítés egyik kulcseleme. Az Ipar 4.0 a hagyományos gépeken alapuló gyártást digitális, intelligens rendszerekre cseréli. A koncepció Németországból indult ki, és célja, hogy az ipari folyamatokat automatizáltabbá, rugalmasabbá és hatékonyabbá tegye a modern technológiák integrálásával [3].

Az Ipar 4.0 létrehozatalának néhány oka:

- Technológiai megoldást keresünk a növekvő üzleti, társadalmi és környezeti problémákra.
- Egyre nagyobb figyelem a vásárlói igényekre, így az egyénre szabott termékek arányának növekedése.
- A fentiekhez kell illeszteni a gyártási folyamatokat és technológiákat, és az ehhez szükséges fejlesztéseket elvégezni.
- Egész iparágakat fog érinteni, mert átalakítja az áruk tervezésének, gyártásának, szállításának és fizetésének a módját.
- Azok a vállalatok lesznek sikeresek a jövőben, amelyek az egyedi vevői igények alacsony költségen való kielégítésére lesznek képesek.

Az Ipar 4.0 technológiák nemcsak kiegészítik egymást, hanem feltételezi is az egyik technológia a másik meglétét, pl.: a „big data” elemzéshez szükséges a dolgok internete a szenzorhálózat miatt, a felhő technológiák az adattárolás és számítás miatt, valamint a megfelelő tudás és szoftvermegoldások rendelkezésre állása. Így alkot az egész koncepció egy magasabb egységet, amelyben megvalósul a gazdaságos, mégis az egyéni igények szerinti gyártás. A koncepció összetettsége magyarázatot ad arra is, hogy miért ilyen lassú a teljes körű bevezetése az iparban.

Az Ipar 4.0 koncepciónak többféle megfogalmazása és megközelítése használatos, ebből válogattam ki néhányat, hogy közelebb kerüljek ehhez a trendhez. Az Ipar 4.0 paradigmája a teljesítményorientált digitalizáció és automatizáció révén a gyártási folyamatok optimalizálására és a termelékenység növelésére fókuszál [4]. A megfogalmazásom szerint: az Ipar 4.0 olyan gyártási szemléletmód, amelyben az okos termékek okos gyárakban készülnek, magasszintű hálózati integráció mellett az egyedi vevői igényeket kielégítve, és az egyes elemek az adott információk alapján szervezik a saját működésüket a többi komponenssel összehangoltan a gazdaságos gyártás érdekében. Kiegészítve továbbá a termelési láncok, az értékláncok, a logisztika és az értékesítés digitalizált csatornáival, amely lefedi a termékek teljes tervezési-gyártási-értékesítési ciklusát. Ez a koncepció egészül ki az Ipar 5.0-ban egy emberközpontú megközelítéssel, ahol a dolgozó, a robotok és az intelligens rendszerek közötti szinergia és kreativitás áll a középpontban. Míg az Ipar 4.0 elsődlegesen a rugalmasságra és hatékonyságra törekedett, addig az Ipar 5.0 a fenntarthatóságot, a jólétet és a társadalmi felelősségvállalást integrálja a gyártási célok közé. A szervezeti struktúrában az Ipar 4.0 decentralizált, moduláris hálózati modelleket alkalmazott, az Ipar 5.0 pedig kiegészíti ezeket kollaboratív innovációs terek és humánközpontú vezetési modellek kialakításával. A technológiai komponensek mindkét paradigma alapját képezik, azonban az Ipar 5.0 esetében hangsúlyosabb a fenntartható és etikus gyártás, valamint a társadalmi kihívásokra adott válaszok. Az Ipar 4.0 korlátokba ütközik a fenntarthatóság és a munkavállalók jóléte szempontjából, mert a digitalizáció révén a termelés rugalmasságára és hatékonyságára összpontosít. Ezzel szemben az Ipar 5.0 célja a munkavállalók jóléte, emberközpontú megoldásokkal

[5]. Más megközelítésben az Ipar 5.0 középpontjában az ember és a gépek közötti együttműködés áll.

Az Ipar 4.0 egyik legfontosabb jellemzője a gyáron belüli különböző komponensek vertikális integrációja egy rugalmas és átkonfigurálható gyártási rendszer, azaz az intelligens gyár megvalósítása érdekében. Wang és társai egy keretrendszert mutatnak be tanulmányukban, amelyben az ipari hálózatot, a felhőt és a felügyeleti vezérlőterminálokat integrálja az olyan intelligens egységekkel, mint a berendezések és szállítószalagok [6]. Autonóm döntés jellemzi, amely nagyfokú rugalmasságot biztosít a rendszernek. Javasolnak egy intelligens tárgyalási módot az ágensek egymással való kommunikációjához, amely javító hatással van többek között a holtponatok megelőzésére is.

Az Ipar 4.0 kulcsfontosságú öt aspektusa az alábbi:

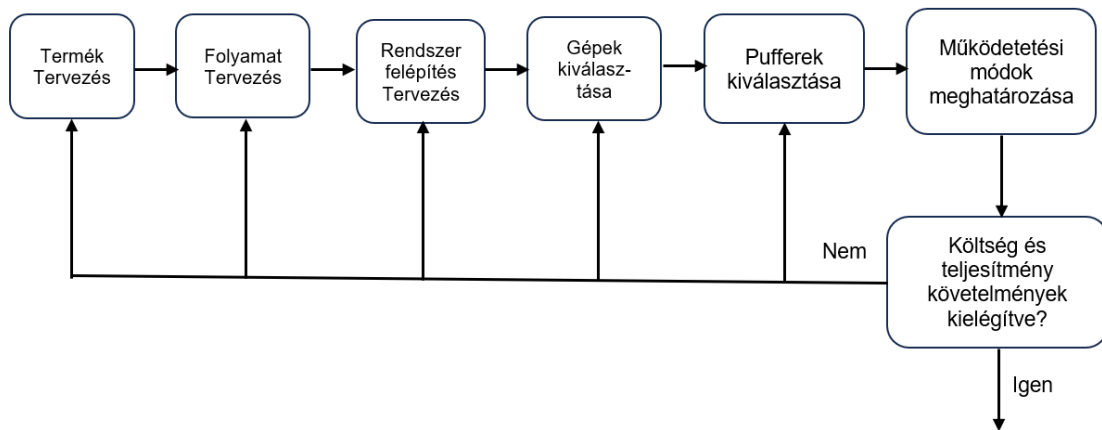
- Kiberfizikai rendszerek (CPS): a fizikailag megjelenő eszközöket köti össze a kibertérrel, adatot termel az adott folyamatról. Képes kölcsönhatásba lépni a környezetével az érzékelőin és aktuátorain keresztül.
- Dolgok internete (IoT): hálózati eszközökkel egyesíti a fizikai érzékelőkből nyert adatokat, és így támogatja a döntéshozatalt. Integrálja a fizikai világot a virtuális világba.
- Szolgáltatások internete (IoS): elsősorban a szolgáltatások webes technológiák általi rugalmas elérhetővé tételéről szól.
- Intelligens gyár (Smart Factory): CPS és IoT technológiák használata, ahol emberek és gépek könnyen tudnak kommunikálni egymással a minél hatékonyabb, egyéni igényeket kielégítő gyártás érdekében.
- Big Data: a nagyméretű strukturálatlan adathalmazok feldolgozása, korábban nem ismert összefüggések felderítése és következtetések levonása érdekében.

A várható társadalmi és gazdasági hatásai úgymint a munkaerőpiac átalakulása, új üzleti modellek, adatalapú szolgáltatások mellett az ember nélküli ún. sötét gyárak vagy az okos városok és a viselhető technológiák megjelenésével, elterjedésével is számolni kell [3]. Az Ipar 4.0 ezek alapján nem csak egy technológiai fejlesztés, hanem egy teljes körű ipari és társadalmi

átalakulás, amely új szemléletet követel meg a vállalatoktól, döntéshozóktól és az oktatási rendszertől egyaránt.

1.2 A gyártórendszerek tervezése

A gyártórendszerek tervezésének témaköre a gyárak tervezésével és működtetésével foglalkozik [7].



1. ábra Gyártórendszer tervezési folyamat

Az 1. ábrán a teljes gyártórendszer tervezési folyamatából a mérnöki tevékenységek láthatóak. A kutatásom szempontjából ez a három lépés emelhető ki:

- A gyár szerkezetének (architektúrájának) tervezése.
- A gépek kiválasztása.

A működtetési mód (működési politika) meghatározza, hogy a termelés során hogyan hozzák meg a döntéseket. A modell szinten felépített gyárat ezek után értékelik, minősítik vagy szimulációval, vagy a korábbi tapasztalatok alapján. Ha az eredmények kielégítőek, akkor a tervezetet elfogadják, és megkezdődhet a kivitelezés. Ha nem kielégítő a teljesítmény, akkor a tervezés valamely lépését és az összes azt követő lépést megismétlik. Addig ismétlik ezt a folyamatot amíg az eredmény kielégítő nem lesz.

1.3 A gyártórendszerek elemzése és szimulációja

A gyártórendszerek elemzésekor ilyen és ehhez hasonló kérdésekre keresik a mérnökök a válaszokat:

- Adott erőforrás kapacitás feltételezése mellett, mekkora lehet egy termékből a maximális kihozatal?
- Milyen termékből mennyit kell gyártani, hogy a haszon a maximális legyen?
- Hogyan lehet úgy elvégezni a gyártást, hogy a környezet terhelése minimális legyen?

A felmerülő kérdések megválaszolásához előzetes tervek és elemzések szükségesek, majd alternatívák felállítása, végük ezek közül választanak a döntéshozók, valamilyen szempont szerint.

Valamely szempont szerinti legjobb állapot meghatározását optimálásnak nevezzük. A modellezés szükséges feltétele a rendszerek formális eszközökkel, vagy szoftverekkel történő kezelésének. Ezek a modellek elsősorban grafok, vagy formális matematikai leírások, vagyis halmazok, változók, és ezeken értelmezett feltételek, műveletek és összefüggések.

Más megfogalmazás szerint, amikor egy gyártási folyamatot módszeresen tervezünk, akkor nagyszámú megvalósítható alternatíva közül kell folyamatosan választanunk. Ha a választások közben törekszünk arra, hogy az alternatívák közül a lehető legjobbat válasszuk, akkor optimálást hajtunk végre.

Minta fenti a leírásból is látszik, alapvetően ezek a modellek a korábbi, előre megadott termékekkel, vagy termék típusokkal kalkuláló gyártáshoz használhatóak elsősorban. Mivel az Ipari 4.0 egyik alapvető célja a vevői igényekre való gyártás, a raktárra történő tömegtermelés helyett, ezért is indokoltnak éreztem más megközelítéseket is kutatni.

Nyembaa és társai egy faáru gyártási folyamat modellezését és szimulációját végezték el, melyekből felírható ilyen modellezési és szimulációs folyamat általános esetekre is [8].

Változó terhelést jelentenek a rendszerre [9]:

- A termelési ütemtervek.

- A termékválaszték változásai.
- Az anyagmozgató berendezések elérhetősége.
- Véletlenszerű meghibásodások.

Ezért is használják mind a statikus mind a dinamikus elemzést az üzemek hatékonyságának értékelésekor az anyagáramlás szempontjából.

A modellezése és a szimuláció általában a probléma megfelelő azonosításával kezdődik, amely magában foglalja a célok meghatározását és a vizsgálandó rendszer releváns szabályozható és nem szabályozható változóinak azonosítását. Minden géphez négy lehetséges állapotot rendeltek: tétlen, foglalt, megtelt és meghibásodott. A meghibásodási állapot a gép átlagos üzem idejének és állás idejének százalékos arányától függött. A szimulációs eredmények megmutatták:

- Egy termék által a rendszerben töltött átlagos időt.
- Az átlagos óránkénti áteresztő képességet.
- A gépek: tétlen, foglalt, megtelt és meghibásodott időtartamát.

1.4 Petri-hálók

A Petri hálók olyan eszközök, amelyek alkalmasak folyamatok szimulációjára.

Egy Petri-hálózat az (1) szerinti módon írható le.

$$PN = (P, T, I^+, I^-, M_0) \quad (1)$$

Ahol:

- P a helyek halmaza:

$$P = \{p_1, p_2, p_3, \dots, p_\pi\} \quad (2)$$

- T az átmenetek halmaza:

$$T = \{t_1, t_2, t_3, \dots, t_\tau\} \quad (3)$$

- I^+ és I^- az éleket leíró mátrixok,
- M_0 pedig a kezdeti állapotot leíró vektor.

Az I^+ és I^- mátrixok τ sorból és π , oszlopból állnak. Az $I^+(i, j)$ megmondja, hogy a t_i átmenetből hány él vezet a p_j hely felé. Az $I^-(i, j)$ megmondja, hogy a p_j helytől hány él vezet a t_i átmenet felé. Mindkét mátrix természetes számokat tartalmaz. A helyeken ún. „tokenek” lehetnek, és a hálózat jellemzője ezen „tokenek” száma. Ezek a „tokenek” valósítják meg az állapotok megváltozását, és ezek eloszlása adja meg a Petri-hálózat egy adott állapotát. Ezen eloszlás kiindulási helyzetét az M_0 vektor adja meg, melynek π számú eleme van.

A helyek az átmenetekhez csatlakoznak, az átmenetek pedig a helyekhez. Ezeket az élek kötik össze.

Az átmenetek valósítják meg az ún. „tüzelést” amikor minden bemenete engedélyezett, vagyis van token az adott helyen. Matematikailag ez a (4) -es egyenlet szerinti megfogalmazás.

$$\forall p_j \in \circ \ t_i I^-(i, j) \leq M(j) \quad (4)$$

Azaz a p_j helyen van legalább annyi token, ahányszoros él vezet a p_j -ből a t_i -be. Az éleknek lehet súlyt is adni, vagyis, ha egynél több token szükséges a tüzeléshez ezt itt lehet megadni.

A Petri-hálóknak az alábbi tulajdonságait tekintem át:

- Korlátosság: a hálózat „ k ” korlátos, ha egyik helyén sincsen több mint „ k ” db token. Vagyis a tokenek száma véges vagy végtelen tüzelés után is véges marad.
- Biztonságos: akkor biztonságos, ha $k=1$, vagyis 1 korlátos.
- Élőség: egy átmenet akkor és csak akkor élő, ha bármely „ n ” természetes szám esetén, egy „ n ” számú tüzelési lépéssorozat után még biztosan tüzelni fog.
 - Az egész hálózat akkor élő, ha minden átmenete élő.
 - Amelyik hálózatban forrás vagy nyelő átmenet van, az nem lehet egyidejűleg élő és korlátos is.
- Az elérhető állapotokat (M vektorok sorozata) gráfban ábrázolhatjuk.
- Elérhetőség: elérhető-e valamely M állapot az M_0 -ból kiindulva?

A klasszikus Petri-hálóknak három gyengesége van [10]:

- Hatalmas hálókat eredményeznek az összetett rendszerek modelljei, ezeket időigényes létrehozni, és nehezen is értelmezhetőek.
- A kifejező erejük korlátozott, nem lehet velük pl.: $x \leq y$ jellegű feltételeket vizsgálni.
- Nem kezelik az időt, ezért teljesítmény mérésére alkalmatlanok.

Ezeket a gondokat a hierarchia hiánya, az azonos színű (fekete) tokenek, és az átmenetek tüzelésének időtől való függetlensége okozzák. Ezért a megoldások is ennek megfelelően:

- Színes tokenekkel bevezethetünk feltételeket.
- Időzítéssel már le lehet írni a rendszer időbeli változásait.
- Hierarchiát is adhatunk a Petri-hálóhoz.

Az így létrejött időzített, hierarchikus és színezett Petri-hálót magas szintű Petri-hálónak szokás nevezni.

1.5 Minősítés szimulációval

A gyártórendszerek célja termékek előállítása, melyeknél szintén használnak tudásbázis alapú megközelítéseket [11].

A flexibilis gyártórendszerek a valós esetekben bonyolultságuk miatt főleg szimulációs eszközökkel vizsgálhatók, modellezhetők [12]. Insup és társai [13] a teljesítmény maximalizálása érdekében számos tényezőt figyelembe vesznek az optimalálásnál, ezek közül néhány: AGV-k száma (Automated Guided Vehicle – automatikusan vezetett jármű), sebesség, ütemezés és puffer méretek. Az így modellezett gyártórendszerben az alábbiak szerint mérik a teljesítményt: 1. minimalizálni a torlódást (congestion), 2. minimalizálni a kihasználtságot (utilization). 3. maximalizálni a kibocsátási teljesítményt (throughput). A kihasználtság minimalizálása arra utal, hogy a szükségesnél többet ne mozogjanak a robotok.

Németh és társai egy beépülő szimulációs modult ismertettek a Plant Simulation szoftverhez [14]. Ezzel komplex elrendezési terveket lehet létrehozni gyorsan sablonok alapján, amelyekben az AGV-k útvonala is konfigurálható. Fedorko és társai a Plant Simulation szoftverben a forgalomcsomópontok

működését elemzik a torlódások jobb megértése érdekében [15]. Malega és társai egy gyártórendszer optimalizálást végeztek el a szoftverrel, többféle beállítási lehetőséget is vizsgálva, és igazolták, hogy a szimulációs eszközök kulcsfontosságúak a teljesítmény növelésében és a költségek csökkentésében [16]. Mások a gyártósor digitalizálását kutatják, és azt, hogy a virtuális modell hogyan tudja támogatni a valós folyamatok elemzését és fejlesztését [17], míg vannak kutatások a termelési üzemek elrendezésének optimalizálására is, és az így ismertetett új módszerrel a termelési és logisztikai folyamatok hatékonysága növelhető [18].

Számos kutató fejleszt és használ saját szimulációt is. Az egyszerű egy robot egy útvonal esetétől [19] a bonyolultabb több robotos modellekig [20] sokféle rendszer felírható és vizsgálható diszkrét eseményű rendszerek alapján is [21]. Mások matematikai modellekkel, egy összetett minimalizálási problémaként írják le a mobil robotot használó gyártórendszert [22]. De vannak kutatások a fenntarthatóság irányában is [23]. Ionescu és társai grafikus programozói felületeket hasonlítottak össze, közös módszertant kidolgozva a programozás egységesítésére [24].

A valós gyártórendszereket általában diszkrét eseményű rendszerekkel (discrete event systems – DES) szimulálják, amelyek felgyorsítva képesek meghatározni a felvázolt gyártórendszerek várható paramétereit, működési határait, optimális működési tartományukat. Számos kutató foglalkozik a gyártás szimulációjával, egyesek pedig az ezekből írt cikkeket elemzik, és vannak lekövetkeztetések. Így tettek Jahangirian és társai is cikkükben, melyben 281 tanulmányt elemeztek és tettek megállapításokat [25]. Az elsősorban a valós alkalmazásokra fókuszáló kutatás az üzleti élet érdeklődését is figyelembe vette, és megállapította, hogy a DES népszerűsége ellenére, nincs meg az üzleti élet bevonásának akkora mértéke (érdeklődése) mint a pl.: a közlekedési szimuláció esetében. Ezt a nehézkes adatgyűjtésnek és az *időigényességnek* tudják be a szerzők, amely tényezők taszítják az üzleti szereplőket. A rendszer dinamikus esetben (system dynamics – SD) az üzleti felek bevonása nagyobb, mivel szabványosított koncepcionális modellezést alkalmaz. Jahangirian tanulmányában az alkalmazási szempontból az ütemezés kapta a legmagasabb

pontszámot, ami érthető, hiszen ez egy jól körül határolható terület, ismert változókkal.

1.6 Gyártórendszerek optimalizálása

Optimalizálással számos helyen találkozunk, az egyszerűbb, hagyományosabb algoritmusoktól a jóval összetettebb, vagy újszerű megközelítést alkalmazókig. Brahami és társai áttekintő cikkében az RMS-ek (Reconfigurable Manufacturing System – újra konfigurálható gyártórendszer) optimalizálását ismertetik [26]. A matematikai programozások fajtáit és alkalmazását láthatjuk az 1. táblázatban.

1. táblázat MCDM módszerek összehasonlítása

	MINLP	MILP	ILP	LP
Leírás	Vegyes egészértékű, nemlineáris változókat is kezelő modell.	Vegyes egészértékű lineáris célfüggvényű modell.	Egészértékű programozás csak 0 vagy 1 változókkal.	Folytonos változójú lineáris célfüggvényű modell.
Alkalmazás	Profit-maximalizálás, optimális termelési sorrend meghatározása, akár kettős célfüggvényt lehet kezelni.	Konfiguráció és ütemezés optimalizálás, gépkiválasztás és költségminimalizálás.	Beruházási költség minimalizálása.	Energiaköltség minimalizálás és átbocsátás maximalizálás.
Előny	Pontos, egzakt megoldást nyújt, jól kezeli a nemlinearitást.	Jól skálázható nagyobb rendszerekre, hatékonyabb, mint a nemlineáris modell.	Jól használható diszkrét döntéseknél, erős matematikai háttér.	Rendkívül gyors, és skálázható; széleskörűen alkalmazható költség és erőforrás-allokációs problémákra.
Hátrány	Nagy számítási igény, csak kis-közepes méretű RMS-ekre skálázható.	Csak akkor használható, ha a probléma lineáris keretbe foglalható, így a valós eseteket linearizálni kell.	Nem kezeli a folytonos változókat, így termelési mennyiségek optimalizálásához korlátozott.	Nem tud diszkrét döntéseket kezelni, noha a valós RMS-ben egészértékű változókra gyakran van szükség.

Az optimalizálási probléma megoldási módszereit így foglaltam össze:

- Matematikai programozás: MINLP, MILP, ILP vagy LP modellekhez pontos formalizálás szükséges.
- Dinamikus programozás: Többperiódusos konfiguráció-útvonalak és áramláskiegyenlítési feladatok optimális rendezésére használatos.
- Egyszerű heurisztikák: gyors, szabály alapú eljárások folyamat és layout tervezéshez.
- Meta – heurisztikák: pl.: genetikus algoritmusok.

Röviden a genetikus algoritmust is ismertetem, mint az optimalizálási feladatok egy elterjedten alkalmazott megoldó eszközét. A működését úgy képzelhetjük el, mintha adva lenne egy feladat, aminek a megoldása felírható egy állapotvektorral, vagyis egy számsorral. Ezt a számsort tudjuk minősíteni egy értékelési kritérium szerint, ez lesz a célfüggvény vagy fitness függvény. A cél egy olyan számsornak az előállítása amelyik célfüggvényének az értéke maximális. A genetikus algoritmust a természetes szelekció ihlette. Működésének fő lépései:

1. Populáció inicializálása: egy véletlen generált egyedekből álló populáció létrehozása, amely 0-1 vagy akár szöveg formában tárolja a döntési változókat (pl.: ütemezési sorrendterv). Ezek lesznek a szülő egyedek.
2. Alakító operátorok: kétféle van. Egyik a keresztezés, amely két szülő információjának (számsor, ami tekinthető a genomnak is) kombinálása révén új egyedeket hoz létre. Második a mutáció, amely véletlenszerűen hajt végre cseréket a genomon.
3. Fitness értékelés: minden egyedet kiértékel a célfüggvény szerint.
4. Szelekció: kiválasztja a leg fittebb egyedeket, amelyek átkerülnek a következő generációba (szülőként). Ennek egy változatában a korábbi szülők legjobb példányait is átörökíti, ezt nevezik „elitizmusnak”.
5. Ciklikusság: a 2 lépéstől a lépések addig ismétlődnek amíg el nem érjük a megadott iterációs korlátot, esetleg időkorlátot, vagy akár az optimális megoldás tetszőleges megközelítését, vagyis a megoldás küszöböt.

Genetikus algoritmusra részletes példát találunk Singh és Sundar cikkében [27], amelyben egy hibrid genetikus algoritmust mutatnak be, amely bit- és

permutációs reprezentációt is alkalmaz az MST-kereséshez (MST – minimum spanning tree, minimális feszítő fa). A populáció inicializálása véletlenszerű feszítőfa-megoldásokkal történik, majd tornaversenyesszelekció, egy- és kéttagú keresztezés, mutáció és elitizmus kombinációjával konvergálnak a minimális összsúlyú feszítőfára. Kísérleteik kis és közepes méretű gráfokon azt mutatják, hogy a GA megközelítés versenyképes a klasszikus Kruskal- és Prim-algoritmussal.

Brahmi [26] három példát is hoz a genetikus algoritmus alapú optimalásra: moduláris termékcsalád ütemezése, beruházási és működési költségek minimalizálása és gyártási sor kiegyenlítése.

Yelles-Chaouche és társai cikkében az RMS -ket vizsgálták az optimalás szempontjából [28], a hangsúlyt a skálázhatóság, költség- és időoptimalási szempontokra helyezve. Néhány optimalizálási célfüggvény gépi szinten: költség, idő, alternatív konfigurációk száma, gépek kihasználtsága. A célfüggvények rendszer szinten: összköltség, teljesítmény, ciklusidő, kapacitáskihasználtság, energiafogyasztás, rugalmasság, megbízhatóság, rendelkezésre állás. Egy lehetőség amire használni lehet, és a kutatás szempontjából fontos: az elrendezési vázlatok variálása, hogy az anyagmozgatási költségeket minimalizálják.

A tetőcserép gyártásban való gyakorlati alkalmazást mutat be Navalertporn és társai tanulmánya [29]. Részecskeraj alapú optimalizációt használnak a többcélú optimalás végrehajtásához, míg neurális hálót a döntési változók és a teljesítmény mutatók közötti összefüggések felderítéséhez. Mások diszkretizálják a mesterséges méhraj kolónia (discrete artificial bee colony – DABC) algoritmust, hogy megoldjanak vele egy ütemezési problémát [30]

1.7 Gyártórendszer minősítéséhez mérőszámok

Az egyes gyártórendszerek teljesítményét mérni is szükséges. A mérésekkel egzakt eredményeket kapunk, és ezáltal össze is hasonlíthatjuk az egyes gyártórendszer konfigurációkat egymással.

A következőkben az egyes különálló eszközök teljesítmény méréséhez használható mérőszámok ismertetésére kerül sor. Az egyik elterjedten használt mérőszám a legyártott jó darabok mennyiségét veti össze az összes adott idő

alatt elméletileg legyártható darabszámmal. Ennek a mutatószámnak a neve magyarul a „teljes eszköz hatékonyság” vagy „általános eszközkivhasználtság”, angolul az OEE (Overall Equipment Effectiveness) [31]. A 2. ábrán látható a gyártási idők egymáshoz viszonyított aránya, vagyis az idők csökkenésének okai és az egyes idők elnevezései.

Az elméletileg gyártásra fordítható idő és a ténylegesen legyártott jó darabok gyártási idejének összefüggése	
Naptári idő (24/7)	Időcsökkenés oka
Tényleges rendelkezésre álló idő	A gyár be van zárva
Tervezett működési idő	Tervezett leállítás
Tényleges működési idő	Nem tervezett leállítás
Hasznos működési idő	Kiseb leállítások, késések
Jó termék gyártási ideje	Hibás gyártás

2. ábra Gyártási idők értelmezése

Az OEE három tényezőtől áll össze:

- Rendelkezésre állás (Availability)
- Teljesítmény (Performance)
- Minőség (Quality)

A rendelkezésre állás (A – Availability mutató) az (5) -ös egyenletben szerepel.

$$\text{Rendelkezésre állás} = \left(\frac{\text{Tényleges működési idő}}{\text{Tervezett működési idő}} \right) * 100\% \quad (5)$$

A teljesítmény (P – Performance mutató) a (6) -os egyenletben szerepel.

$$\text{Teljesítmény} = \left(\frac{\text{Hasznos működési idő}}{\text{Tényleges működési idő}} \right) * 100\% \quad (6)$$

A minőség (Q – Quality mutató) a (7) -es egyenletben látható.

$$\text{Minőség} = \left(\frac{\text{Jó termék gyártási ideje}}{\text{Hasznos működési idő}} \right) * 100\% \quad (7)$$

Az OEE-t a (8) -as egyenlet szerint lehet kiszámolni, vagyis az előző három mutató szorzata.

$$OEE = A * P * Q \quad (8)$$

Kifejtve a (9) -es egyenletben látható.

$$OEE = \left(\frac{Tényleges\ működési\ idő}{Tervezett\ működési\ idő} * \frac{Hasznos\ működési\ idő}{Tényleges\ működési\ idő} * \frac{Jó\ termék\ gyártási\ ideje}{Hasznos\ működési\ idő} \right) * 100\% \quad (9)$$

Egyszerűsítve pedig a (10) -es egyenletben látható.

$$OEE = \left(\frac{Jó\ termék\ gyártási\ ideje}{Tervezett\ működési\ idő} \right) * 100\% \quad (10)$$

A magasabb szintű hatékonyságok mérésére is vannak mutatószámok. Az egyik ilyen egy gyár szintű mutató az általános üzemhatékonyság (Overall Plant Effectiveness – OPE)

A gyártósorra vonatkozó mutató az „általános sor hatékonyság” (Overall Line Effectiveness – OLE).

Szintén használatos még a „gyártósori általános eszközhatékonyság” mutató (Overall Equipment Effectiveness of a Manufacturing Line – OEEML)

Más mutatók a teljesítményt vagy a minőséget a legyártott termékek darabszámához is köthetik.

1.8 A gyártás minősítéséhez mérőszámok

Az alábbiakban Afteni és társa cikke alapján ismertetem a gyártásban használatos minőségi mérőszámokat [32]. A jól összefoglalt tanulmányban számos újfajta mérőszámot és optimalizációs lehetőséget ismertetnek. Az optimalizációs feladatot úgy értelmezik, mint egy tevékenység, amelynek során egy probléma lehetséges megoldásai közül kiválasztjuk a legjobbat, amely megfelel egy előre meghatározott kritériumnak. Három komponenst feltételez:

- A célfüggvény, aminek keressük a szélső értékét:

$$\min, \max f(X), \quad x \in R^m \quad (11)$$

- A változók vektora, X -el jelölve:

$$X = (x_1, x_2, \dots, x_n) \quad (12)$$

- Valamint a korlátozások az alábbi formákban, az egyenlőtlenségi korlát a (13) -as egyenletben, míg az egyenlőségi korlát a (14) -es egyenletben láthatóak.

$$g(X) \leq 0 \text{ – az egyenlőtlenségi korlát} \quad (13)$$

$$h(X) = 0 \text{ – az egyenlőségi korlát} \quad (14)$$

Az alábbi teljesítmény mérésére alkalmas mérőszámok elsősorban a cikkben a forgácsolásra vonatkoznak, de úgy gondolom, hogy egy részük általánosítható más területekre, így a gyártórendszerek teljesítményének mérésére is.

- A termelékenységet a (15) -ös képletben látható.

$$Q = v * s * t \quad (15)$$

vagy, másképp felírva a (16) -os egyenletben.

$$Q = \frac{k}{T_m} \quad (16)$$

○ Ahol:

- v: forgácsolási sebesség
- s: előtolási sebesség
- t: fogásmélység
- k: konstans
- T_m: a gyártási időt jelenti.

- Költséget az alábbi formában írja fel:

$$C_z = \sum_{i=1}^n \left(\frac{Q_i}{T_i} \right) * \left[1 + k + \frac{\tau_{sr}}{T} \right] * \frac{1}{v * s * t} + \frac{\tau_{sr} * c_{\tau} * c_s}{T * v * s * t} + \frac{c_{\tau}}{v * s * t} \quad (17)$$

$$+ c_{mater} + \frac{k_{energy} * c_{energy}}{v * s * t}, \left(\frac{Euro}{cm^3} \right)$$

○ Ahol:

- Q_i: az „i”-edik eszköz értéke, az össze feldolgozott mennyiségére
- T_i: az „i”-edik eszköz élettartama

- c_T : bérfajlagos költség
- c_S : a szerszámra fordított kiadás két egymást követő szerszámcsere között
- τ_{SR} : a kopott szerszám cseréjének ideje
- c_{mater} : a leválasztott anyag költsége
- k_{energy} : az energiatényező
- c_{energy} : az energia ára
- Amivel a (17) -es egyenlet kiszámolt értéke csökkenthető: olcsóbb munkaerő, alacsonyabb energiafogyasztás, szerszám és alapanyag költségének csökkentése
- Az energiafogyasztás:

$$EC_w = T_w * \left(a * \frac{K_w}{T_w} + b \right)^{-1}, \left(\frac{Kwh}{m\ddot{u}velet} \right) \quad (18)$$

- Ahol:
 - T_w : egy művelet munkaideje
 - K_w : arányossági állandó
 - a és b : fajlagos állandók
- Cél a veszteségek elkerülése, költségmegtakarítás, a környezetre gyakorolt hatás csökkentése
- Keresőképesség a (19) szerint van deklarálnva.

$$EP_{ijk} = \frac{P_{ijk} - c_{ijk}(p_{ijk})}{A_{ijk} - t_{ijk}(p_{ijk})}, \left(\frac{Euro}{Euro * perc} \right) \quad (19)$$

- Ahol:
 - P_{ijk} : az ár,
 - $c_{ijk}(p_{ijk})$: az adott megfelelő költségek
 - p_{ijk} : a paramétervektor függvényei
 - A_{ijk} : A végrehajtó munkaállomás eszköze
 - $t_{ijk}(p_{ijk})$: a köteget művelet ideje, amíg az „i”-edik rendeléshez tartozó „k”-adik műveletet végzi a „j” munkán.

- Profit ráta:

$$PR = \frac{P_s - C_s}{\left(1 + k + \frac{\tau_{sr}}{T}\right) * \frac{1}{v * s * t}}, \left(\frac{Euro}{perc}\right) \quad (20)$$

- Ahol:

- P_s : a P/V fajlagos ár aránya
- C_s : a C/V fajlagos költség aránya
- τ_{sr} : a kopott szerszám cseréjének ideje, percben.

- Befektetés hatékonysága egy pénzügyi arányszámként szerepel a (21)-ben.

$$IE = \frac{P_s - C_s}{\sum_{i=1}^n Q_i}, \left(\frac{Euro}{Euro}\right) \quad (21)$$

- Ahol a Q_i : az „i”-edik eszköz értékét jelenti az n-ből, amely a folyamat működéséhez szükséges.

- Fenntartható profit: ennek képletében a szerszám gyártásával járó szén-dioxid kibocsátás, a fajlagos energiafogyasztás, és a megmunkálás során keletkező fajlagos szén-dioxid kibocsátás szerepel. Mértékegysége: (Euro/kg CO₂)

Afteni és társai [32] tanulmányuk végén megemlítik, hogy az optimálásnak vagy több kritériumosnak kell lennie, vagy sok célfüggvényre külön-külön kell elvégezni, hogy hatékony döntéstámogató eszköz legyen a gyártási folyamat menedzseléséhez.

1.9 A gyártási környezet várható jövőbeli trendjei

Számos kutatás foglalkozik a jövő gyarával, felépítésével, működésével [33]. Sokféle elképzelés létezik, de abban közösek, hogy a gyártás rugalmassága, az egyedi igényekhez való gyors alkalmazkodás kulcskérdés lesz. Egy átfogó ideát mutatnak be Monostori és társai „A platform alapú gyártás” cikkükben. A platform alapú gyártás szerint a vállalatok nem tudják kik készíti az alkatrészeket, a gyártók nem rendelkeznek saját gépekkel, a döntések pedig a cégek határain túl mutatnak. Az alapelve, hogy a tervezett terméket feltöltik egy platformra, és egy nem ismert gyártó előállítja, majd a logisztikai szolgáltató kiszállítja, rövid időn belül a kívánt minőségben és mennyiségben. A jelenlegi nagy „piacterekhez”

képest az a különbség, hogy a terméket nem katalógusból választják ki, hanem igény szerint gyártják. A vállalatok nem tulajdonosok, hanem bérlik a gyártói kapacitást és a szükséges szoftvereket vagy szolgáltatásként fizetnek érte.

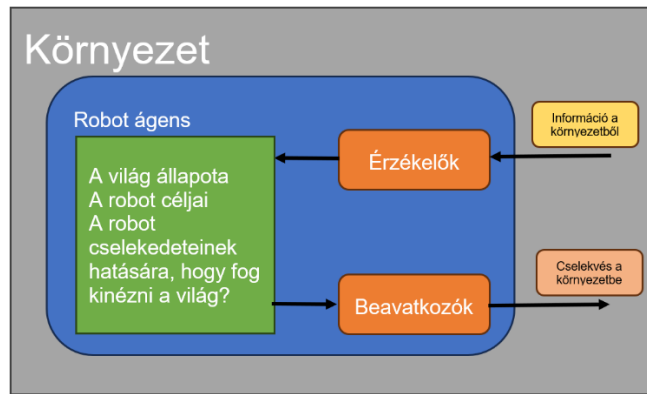
ElMaraghy és társai [34] elemző tanulmányukban az alábbi négy tengely mentén vizsgálják a változásokat: termékek, technológia, üzleti stratégiák és termelési paradigmák. Egyik felvetésük, hogy a robotok egyes problémái miatt (érzékelők korlátozottsága a fényviszonyoktól, valós idejű tárgy érzékelés, hálózati problémák) a jövőben a mesterséges intelligencia jövője a robotikában: az AI alkalmazások kiegészítve a kreatív emberi operátorokkal.

1.10 Mobil robotok

A robot kifejezés Karel Capek 1921-es színdarabjából származik. A műszaki és szépirodalmi fogalomkörben általában olyan eszközt vagy berendezést értenek alatta, amely az ember fizikai és/vagy szellemi tevékenységéhez hasonló munkát végez [35]. Vagyis az ezekből származó *prekonceptiók* alapján a gyártási folyamatokban és a kutatásokban használt robotoktól olyan képességeket várunk el, amellyel a saját tevékenységét ellenőrizni és korrigálni is tudja. Ez azt is jelenti, hogy intelligens viselkedést várunk el tőlük, legalábbis valamilyen szinten. Ennek összetevőit egyszerűen megfogalmazva az alábbiaknak tekintjük:

- Érzékelés.
- Információ-feldolgozó kapacitás.
- Tudás ill. emlékezet.
- Tanulási képesség.
- Döntéseken alapuló közlési-cselekvési képesség.

A szabványokban megtalálható definíciók alapján a robot egy rugalmasan újra programozható, több szabadságfokú mozgási lehetőséggel rendelkező, darabok mozgatására vagy különböző feladatok megvalósítására tervezett eszköz. A 3. ábrán egy sematikus kép látható a robotról, mint ágensről, amely egy adott környezetbe van elhelyezve, amit az érzékelőivel érzékel, majd az ágens a világ állapotát leírva a lehetséges cselekedetinek a hatását értékeli az ágens céljainak a függvényében, és így választja ki a végrehajtandó cselekedetet.



3. ábra Robot ágens sematikus képe

A fenti megfogalmazás a célorientált ágensre érvényes, amelyek az iparban is jellemzőek. Ettől eltérő megfogalmazások is vannak tágítva vagy szűkítve a fenti fogalmat, de a lényegét kiemelve: a robot egy fizikai eszköz, amely programozás után automatikusan képes manipulálni más tárgyakat.

A robotika fejlődésnek indulása a XX. század közepére tehető. Az első robot az Unimate melyet az 1960-as évek elején készítettek gépek kiszolgálására. A robotok fejlődését is generációkba szokták szervezni, ezeket röviden így foglalom össze:

Első generációs: 1960-as években, „felemelés és lerakás” típusú (anyagmozgatási) feladatokra használták. Az esetleges külső érzékelők a védelmet biztosították, de nem voltak jellemzőek. Alacsony programozhatóság, a program a robot mozdulatait egyértelműen leírta.

Második generációs robotok: 1970-es években jelentek meg, ezek már érzékelték a környezetüket, és tudták is módosítani a tevékenységüket a vett jelek alapján. Magasabb szinten programozhatók, megjelennek a robotprogramozási célnyelvek.

Harmadik generációs robotokban már megjelennek a mesterséges intelligencia elemei. Az érzékelőktől származó jeleket feldolgozzák, és a tárolt modell alapján intelligens módon tudnak reagálni, módosítani a pályát, kombinálni az információkat. Megjelennek a döntési rendszerek, magas szintű feladatok fogalmazhatóak meg általánosan is.

Az iparban főleg a második generációs robotokkal találkozhatunk.

1.11 Ipari mobil robotok

Shneier és Bostelman 2015-ben írt átfogó cikkében röviden, mégis sokrétűen bemutatja a mobil robotok világát [36]. Egyszerű és velős megfogalmazásai miatt, jelen disszertációmban több elemet átemeltem. A cikk szerint a mobil robotok olyan eszközök, amelyek autonóm módon mozognak, hogy céljaikat elérjék. Megemlítenek néhány szempontot ahhoz, hogy a mobil robotok alkalmazását segítsék. Az önvezető járműveket (automated guided vehicle – AGV) 1953-ban találták fel. Három alaptípusát különböztethetjük meg: a kocsikat vontató robotokat, az egységtrakományt szállító robotokat és az automatikus villás targoncákat.

A robotok bevezetése viszonylag drága, telepítésük és üzemeltetésük szakértelmet igényel, ezért is nagyon fontos, hogy az adott alkalmazásokban a megfelelő robotokat használjuk, vagyis a kiválasztásuk optimális legyen. A disszertáció célja az optimális kiválasztás elősegítése egy döntéstámogató rendszer fejlesztésével.

Egy tipikus ipari gyártási környezetben több jármű mozoghat, anyagokat vagy alkatrészeket szállítva a munkaállomások között. Ezen járművek működésének számos aspektusát meg kell tervezni, úgymint: ne keresztezzék egymás útvonalát a járművek, ne torlódjon fel a forgalom, az anyagok a megfelelő időben kerüljenek kiszállításra a megfelelő helyre, az áteresztőképesség kompatibilis legyen a gyár munkatempójával, valamint a járművek akkumulátorainak töltésére is legyen idő. A tervezés túlmutat az útvonalak meghatározásán, mert az is benne van, hogy a robotok elkerüljék egymást, az egyéb berendezéseket és a többi embert, ugyanakkor pontosan tudják pozícionálni magukat az egyes állomásokhoz vagy szállítoszalagokhoz.

Shneier és Bostelman [36] cikke megemlíti egy olyan lehetőséget is, hogy (egyéb feltételek teljesülése után) a szerelősorok elrendezése rugalmasabb lesz, akár dinamikusan is átkonfigurálható a termékmix változásának figyelembevételével. *Jelen kutatás szempontjából a cikk legfontosabb mondanivalója, hogy a vásárlóknak szükségük lesz olyan módszerekre, amikkel összehasonlíthatják a termékeket, és meghatározhatják, hogy melyek felnek meg leginkább az igényeiknek.* Ehhez újfajta teljesítménymutatókra és eljárásokra lesz szükség. A következő lépés pedig olyan robotok fejlesztése,

amelyek strukturálatlan gyártókörnyezetben is képesek lokalizálni magukat, megtervezni az útvonalukat, és a manipulátorukkal embereknek szánt feladatokat is elvégezni.

Csernevák Ákos és Husár Jozef tanulmányukban bemutatnak egy multidiszciplináris tanítási modellt a mobil robotok használatához az Ipar 4.0 és 5.0 oktatásához [37]. A tanulási tervben hangsúlyos helyet kap az AGV-k és AMR-ek (Automated Mobile Robot) működésének megértése és programozása.

Az AGV és AMR közötti különbségeket az alábbiakban szerint fogalmazom meg: az AGV-k inkább robotikus/automatikus eszköznek tekintjük, mivel hiányzik belőlük a tényleges autonómia, vagyis, hogy eltérjenek a már előre programozott útvonaltól. Az AMR ezzel szemben önálló navigálásra képes, külső beavatkozás nélkül, tehát ha szükséges akkor dinamikusan újra tervezi az útvonalát. Ugyanakkor megjegyzem, hogy a határvonal a kétféle robot között nem élesen, hanem inkább átfedésekkel értelmezhető, valamint az AMR-eket a környezettől függően lehet AGV-ként is működtetni, míg fordítva ez nem mondható el. Fejlett vezérlő rendszerüknek és tanulási képességüknek köszönhetően az AMR-ek a feladatok végrehajtásában magasabb fokú autonómiát és adaptálódást tesznek lehetővé [38].

Általánosságban elmondható, hogy a robotok integrálása a gyártási folyamatba a következőket teheti szükségessé: műszaki képzések, stratégiai tervezés és a gyártási folyamat átszervezése, valamint a folyamatos innováció [39]. Figyelembe kell venni, különösen az Ipar 5.0 alkalmazásokban az ember-robot interakciót és kommunikációt, valamint az együttműködés lehetőségeit [40]. Ugyanakkor a cégek részéről is van igény olyan rendszerek integrálására amelyek a raktárakból képesek alapanyagot kiszállítani emberi sofőr nélkül [41].

Tovább lehet finomítani az egyes fogalmak közötti különbségeket, amennyiben megkülönböztetjük az automatikus targoncákat az autonóm raklap mozgó robotoktól, amint azt részletesen elemzik ezen az oldalon: <https://www.antdriven.com/what-is-an-automated-forklift>. Míg az utóbbit kifejezetten az autonóm működésre tervezték, addig az előbbit automatikus üzemre képesre alakították ki egy targoncából.

Mobil robotoknál a megoldandó feladatok általában az alábbi három terület köré gyűjthetők: navigálás, útvonaltervezés és forgalomszervezés [42]. A robotok kiválasztását és minősítését szimulációs rendszerekkel is támogatják, ezek összehasonlítása és minősítése szintén kutatások tárgya [43].

Beltéri navigációval számos kutatás foglalkozik, ezeknek közös jellemzője, hogy valamilyen SLAM (Simultaneous Localization and Mapping – lényege: a térképkészítés és a helymeghatározás párhuzamosan történik) rendszerből kiindulva igyekeznek fejleszteni egy speciális feladatra egy jobb megközelítést [44]. Felhasználnak gráf- és mintaalapú algoritmusokat, valamint meta-heurisztikákat, úgymint genetikus, hangyaboly és részecske szimuláció alapú megközelítéseket, esetleg szűrőket, pl.: Kálmán-szűrő [45].

Egyes kutatók a raj-robotika területén végeznek kutatásokat akár formációba rendezésről van szó [46], akár újszerű algoritmusok használatról optimálási feladatokhoz. Kutatási szempontból a két leginkább újszerű megközelítés a mesterséges növényi szaporodásra épülő [47], és a halrajok viselkedésére alapuló megközelítések [48].

Még a mélytanulás fókuszba kerülése előtt, már 2013-ban is a megerősítéses tanulást helyezték kutatásuk homlokterébe Kormushev és társai [49]. Az általuk jegyzett cikkben három problémára alkalmazták a megerősítéses tanulást, többek között az alábbi szempontokat figyelembe véve a reprezentációban: korrelációk, alkalmazkodóképesség, globális és többdimenziós jelleg, valamint konvergencia.

Egyéb területeken is aktívan kutatják a robotok felhasználási lehetőségeit, ezen nem gyáripari alkalmazások gyűjtőneve az ún. szerviz robotika. A teljesség igénye nélkül ide tartoznak a következők: építőipar [50] [51], személyszállítás [52], az egészségügyi szolgáltatások [53], vagy akár egy lakásban megvalósuló otthoni felhasználás [54].

Nem kerülhető meg a robotika tárgyalásakor a mezőgazdasági robotok területe, mert jelenleg ez az egyik olyan terület, ahol a robotok (valamint fejlesztőik és alkalmazóik) azonnal a gyakorlati problémákkal szembesülnek. Röviden ezekből a trendekről is bemutatunk néhányat. Luis Emmi és társai cikkében két részre osztja az automatikus működést, az egyik része az autonóm

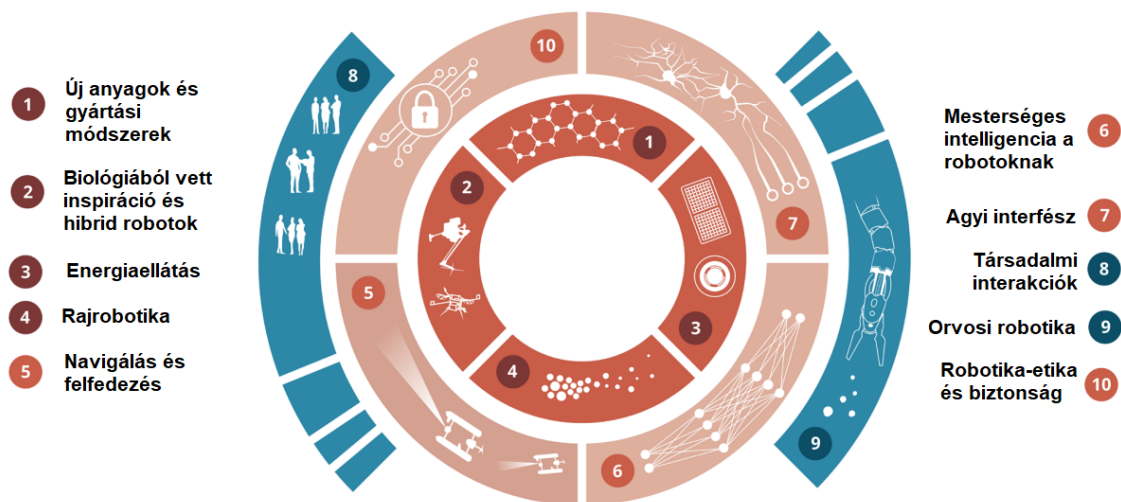
jármű, a másik az autonóm megvalósítás (vehicle és implement) [55]. Az autonóm jármű rész felelős az autonóm irányításért. Ide tartoznak többek között: az érzékelés és gépi látás szenzorai, a GPS, a vezeték nélküli kommunikáció, a közvetlen járműirányítást megvalósító elektromos és hidraulikus aktuátor rendszer. Az autonóm megvalósításba a magasabb szintű feladatok tartoznak, úgymint az autonóm növénytermesztés és flotta irányítás.

Javier és társai cikkében egy kisméretű robotokból álló „kijelzőt” készítettek, ahol a robotok a hagyományos fix osztású pixel alapú kijelzőkkel ellentétben az ő kijelzőjük mozgatható és vezérelhető színeket megjeleníteni képes robotokból épül fel [56]. Egy nagyméretű (akár térbeli) ilyen robotrendszer felépítése és programozása bonyolult feladat, ami nagy szaktudást és tapasztalatot feltételez, ezért az általuk készített kisebb rendszer jó gyakorló és kutató terep a raj-robotika algoritmusok számára. Megoldandó feladatok többek között az ütközésmentes folyamatos mozgás, valamint a vetített képek zavaró átmenet nélküli (rezgésmentes) megjelenítése. A fejlesztett algoritmusuk a cikk szerint skálázható nagyszámú és különféle kinematikájú robotra is. Egyik eredményük, hogy a robotok számát is képesek optimalni a megjelenítendő kép szükséges részleteinek (minőség, felbontás) megfelelően.

A robotok fizikai integrálása a gyártórendszerekbe egy kifejezetten gépészmérnöki kérdés, amiben a mérnökök nagyban támaszkodnak az egyedi gyártás lehetőségeire. Itt a legújabb trendek szerint egyre nagyobb szerepet kap az additív gyártás, vagy egyszerűbb formája a 3D nyomtatást is [57], [58].

1.12 Kihívások a robotika különböző területein

Az eddigiekből is látható, hogy a robotika az egyértelmű előnyei mellett számos kérdést is felvet. Ezeket járják körül Yang és társai a tanulmányukban, megnevezve és részletesen bemutatva a következő 5 – 10 év legfontosabb 10 kihívását [59]. A 4. áttekintő ábrán láthatóak ezek a kihívások.



4. ábra 10 kihívás a robotika területén

Jelent kutatás szempontjából az alábbiak relevánsak:

- Energiaellátás.
- Navigálás és felfedezés.
- Mesterséges intelligencia a robotoknak.

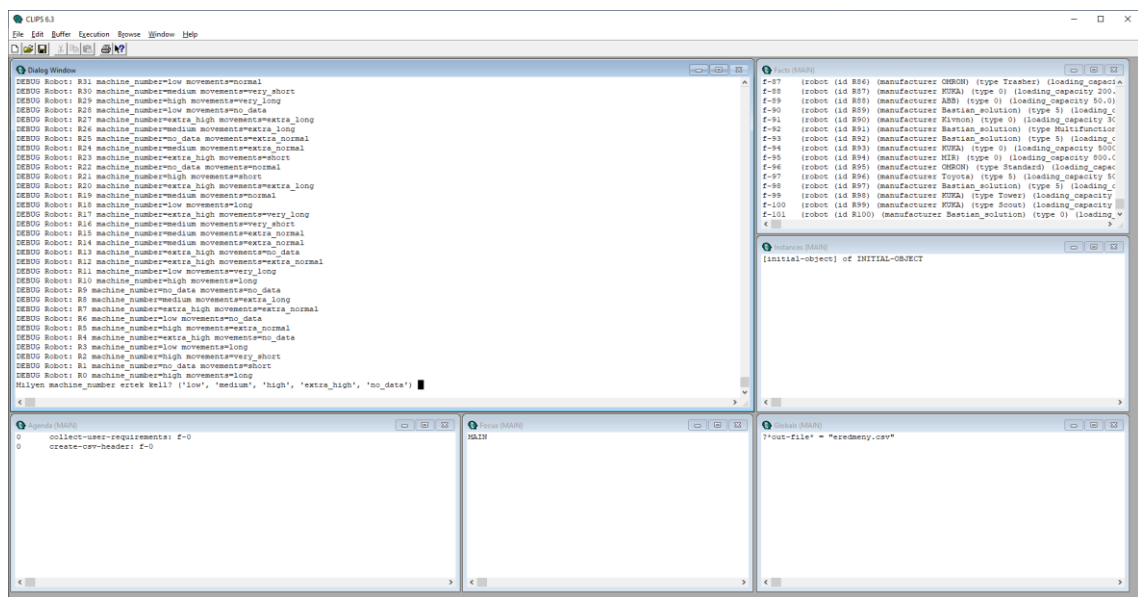
Az energiaellátás a robotok gyártórendszerben való optimális kihasználásának egyik kulcs mozzanata lehet. *Érdeemes elemezni, hogy mekkora akkumulátor kapacitást lehet optimálisan kihasználni a szállítási feladatokra.* A navigálás elsősorban az ismert ipari környezetben való áruszállítást jelenti úgy, hogy a felmerülő dinamikus környezeti változások (pl.: ideiglenes akadály az útvonalon) a robot számára ne jelentsenek problémát a feladat teljesítése szempontjából. Vagyis a robot képes legyen akkor is ellátni a feladatát, ha éppen valamilyen akadály került a pályára, amit ki kell kerülnie, és emiatt egy másik útvonalat kell választania. A mesterséges intelligencia nemcsak a robotok működésében, hanem azok kiválasztásában is segíthet.

1.13 Mesterséges intelligencia, szakértői rendszerek

Tömör de jó összefoglalót találunk Tan cikkében a szakértői rendszerekről [60]. Négyféle szakértői rendszert ismertet: a szabály alapút, a keretrendszer alapút, a fuzzy logikára épülőt, és a neurális hálózatokkal megvalósított szakértői rendszert. Alapvető struktúrája ezeknek a rendszereknek a következő: a tudásbázisból a munkamemóriában a következtető gép elvégzi a műveleteket, így az eredmény az interfészen megjeleníthető. A kutatók a szakértői rendszerek fejlesztése során felismerték, hogy a tudás reprezentáció, a tudás hasznosítása

és a tudás megszerzése a mesterséges szakértői rendszerek három alapvető kérdése.

A disszertációmban a CLIPS szakértői keretrendszert is használtam. Ennek röviden az a lényege, hogy egy teljes szakértői rendszert tudunk vele létrehozni, a fejlesztést csak a szakértői elemekre fókuszálva, mert maga a keret adott, ami az 5. ábrán látható felületen érhető el. A felületen lehet új programot írni, vagy az előre megírt scripteket is be lehet olvasatni, futtatni, majd a megjelenő további ablakokban láthatóak a tények, szabályok, és a futás eredménye.



5. ábra A CLIPS szakértői keretrendszer képe

Ezek a szakértői elemek pedig a tények, és a szabályok. A következőtő gépet a keretrendszer biztosítja.

A tényeket az alábbi formában szükséges megfogalmazni:

- *(assert (fact(...)))* – új tény hozzáadása

Más hasznos parancsok:

- *(printout t ...)* – kiírás
- *(retract ?f)* – tény törlése
- *(modify ?f (slotnév újérték))* – tény módosítása
- *(bind ?változó érték)* – változó értékadás

Gyakoribb megoldás, ha először a tényekhez tartozó sablont adjuk meg, majd ehhez rendeljük az egyes tényekhez tartozó leírásokat. Ezt így kell megtenni:

```
(deftemplate robot
  (slot id)
  (slot status (default idle))
  (slot battery (type number) (default 100))
  (slot carrying-capacity (type number) (default 50))
)
```

Ehhez egy tény így néz ki:

```
(robot (id R1) (status idle) (battery 80) (location A) (carrying-capacity 50))
```

A szabályok sablonja pedig így néz ki általános esetben:

```
(defrule szabály-neve
  (feltétel1)
  (feltétel2)
  =>
  (akció1)
  (akció2)
)
```

Változókra hivatkozni a „?” karakterrel lehet:

- *?változónév* – egyetlen értéket köt le
- *?\$változónév* – több értéket köt le

pl.:

```
(robot (id ?id) (load ?szint))
```

```
(robot (id R1) (load 5))
```

 esetében: *?id* = R1 és *?szint* = 5 lesz.

- *?f <- (...)* – egy teljes fact -ot leköt egy változóhoz, így lehet rá később hivatkozni, pl.: törölni a *retract*-al.
- *?v&:expr* – feltétel egy értékre, pl.:

```
(battery ?b&:(> ?b 50))
```
- több feltételt egymás után kell leírni

Ezek alapján egy összetettebb feladatkiosztáshoz tartozó szabály így néz a CLIPS-ben:

```
(defrule assign-by-skill
  ?t <- (task (id ?tid)
           (type ?skill))
```

```

        (required-level ?req)
        (status pending))
?r <- (robot (id ?rid)
        (status idle)
        (battery ?b&:(> ?b 30))
        (?skill ?level&:(>= ?level ?req)))
=>
(modify ?t (status assigned) (assigned-to ?rid))
(modify ?r (status working))
(printout t "[SKILL ASSIGN] " ?tid " feladat kiosztva robotnak " ?rid
        " (" ?skill " szint: " ?level ")") crlf)
)

```

Ezzel egy robothoz egy feladatot tudunk rendelni úgy, hogy ellenőrizzük a robot töltöttségét, elérhető (munkára kész) állapotát, valamint az adott feladathoz szükséges kompetencia szintjét. Ezeket az ismereteket a CLIPS-ben való munkámhoz használtam fel. Az irodalmi áttekintést az alábbiakban további cikkek bemutatásával folytatom.

A tudásmenedzsment célja, hogy kihasználja a szervezet szakértelmét és tapasztalatát a döntéshozatal javítása, a működés korszerűsítése és a versenyelőny növelése érdekében [61]. Ennek része az együttműködések elősegítése és az információs rendszerek bevezetése mellett a folyamatos tanulás és tudásmegosztás kultúrájának ösztönzése is. A tudás hatékony kezelésével a szervezetek növelhetik a termelékenységet, csökkenthetik a redundanciát, és gyorsabban alkalmazkodhatnak a környezeti változásokhoz. Ehhez a vállaltoknak hozzá kell férniük nagy mennyiségű adathoz minden részlegüktől, majd ezeket felhasználhatják tudásalapú (KBS) vagy szakértői rendszerekben (ES) a folyamatmérnöki szakemberek munkájának segítésére.

Mabkhot és társai egy olyan döntéstámogató keretrendszert mutatnak be, amely a rugalmasan alakítható gyártórendszerek, az RMS-ek újra konfigurálási döntéseit segíti a gyártás közbeni váratlan zavarok vagy igényváltozások esetén [62]. Három megközelítést javasol, egy intuitív súlyozást, egy módosított TOPSIS-t, és az ELECTRE IV-t ami nem igényel döntéshozói beavatkozást. Egy bűvárszivattyú összeszereléssel kapcsolatos esettanulmányt is készítettek, amely

igazolja, hogy a rendszer rugalmasan alkalmazható és támogatja az optimális konfiguráció kiválasztását a döntéshozói részvétel mértékétől függetlenül.

Repülőgépek összeszerelésével kapcsolatban végeztek kutatásokat Szejka és társai [63]. A felállított modelljükben négyféle nézetbe strukturálták a rendszer architektúrát:

- a referencia nézet, a fogalmakat logikai formalizmusba szervezi,
- a szemantikai nézet, az információk kapcsolatát írja le,
- az alkalmazási tartomány, egy tudásmodellbe szervezi a termék vagy gyártás információit,
- az információs nézet, a valós gyártási körülmények.

A rendszer erőssége, hogy hatékonyan és korán azonosítja az információs inkonzisztenciát, hozzájárulva ezzel a proaktív probléma javításhoz, ami egy 700.000 alkatrészt tartalmazó repülőgépnél a versenyképesség mellett a biztonság garantálásában is kiemelten fontos.

Ritou és társai egy tudásbázis alapú és többszintű megközelítést használtak megmunkáláskor keletkező adatok bányászatára és azok kezelésére [64]. A Big Data problémája a sok adat kezelése, ezen segíthetünk azzal, ha az adatokat valós időben feldolgozzuk és így jóval kisebb mennyiségű, de intelligens adatok kerülnek továbbításra a döntéstámogató eszközök felé. A megmunkálási folyamatra vonatkozó kis mennyiségű értelmezett információt használnak fel a döntést segítő mutatók kialakításához a szakértői szándéknak és az üzleti kontextusnak megfelelően.

Al-Mahturi és társai egy type-2 fuzzy rendszert használtak a mobil robotok irányítására nagy bizonytalanságú környezetben [65]. A type-1 fuzzy logikai vezérlő a mobil robotok sajátos viselkedését míg a type-2 a bizonytalan környezethez való alkalmazkodást biztosítja. A szerzők sikeresen implementálták a type-2-be az öntanuló algoritmust csökkentve így a számítási komplexitást és a memória igényt, eljutva ezáltal egy optimalizált fuzzy vezérlő rendszerhez.

Mivel a kinyert adatok nem mindig értelmezhetőek a gép számára, vagy olyan adattárolókban vannak tárolva, amelyek között ugyan van összefüggés, de ez nem jelenik meg, ezért továbbra is szükség van a formális tudásreprezentációs

módszerekre, hogy segítsük a tudás megértését és kiaknázását. Cao és társai olyan hibrid rendszert fejlesztettek és teszteltek, amely a statisztikai és a szimbolikus mesterséges intelligencia eszközöket egyaránt felhasználja a karbantartás előrejelzéséhez [66]. A meghibásodások időrendi sorrendjének segítségével létrehozott szabályok által teremt kapcsolatot az előzmény és a következmény között.

A modern gyártási környezetben a keletkező óriási mennyiségű adat elemzéséhez használt gépi tanulás alapú modellek előre jelző képessége jobb, viszont az átláthatósága rosszabb, mint a hagyományos tudás- vagy modell vezérelt módszereké. Olyan kutatásokról ad átfogó képet Vuković és társai áttekintő cikke, amelyekben az ok-okozati felfedezés gyártásban történő alkalmazását tekintették át [67]. A beazonosított kulcsfontosságú területek: hibák felderítése, elemzése és kezelése; gyökér okok elemzése, az ok-okozati összefüggések közvetítő szerepének elemzése; valamint a területi és a koncepcionális munka. Jelenleg az ok-okozati összefüggések felhasználása az igény szerinti, esetlegesség szintjén vannak, ezt kívánta a cikk fejleszteni a bemutatott módszerekkel, valamint egy útmutatóval.

Imran és társai egy tudásbázis alapú virtuális modellező és szimulációs keretrendszert írtak le az ipar 4.0-hoz kapcsolódóan [68]. Szisztematikusan szedték össze az ipar 4.0 jellemzőit, technikai kihívásait, és bemutattak különféle eseteket is. A technikai kihívások között szerepel a nagy adatmennyiségek feldolgozása a döntések támogatásában. Vagyis az adatok analízisa és értelmezése területén új kutatásokra és az elért eredmények alkalmazására van szükség. A cikkben az adatok valós idejű elkapását (capture) vizsgálták az objektumok a folyamat és a gyár szintjein. Az esettanulmány bemutatta ezen megszerzett adatok valós idejű analízisát és vizualizációját, segítve ezzel a rejtett összefüggések megtalálásában a mérnököket.

A gyártórendszerek jövőbe mutató fejlesztései közül kiemelem Janícek és társai munkáját, amelyben egy kollaboratív robot irányítását oldották meg hangvezérléssel [69]. Ez a vezérlési mód egyre népszerűbb, elég csak az okos otthonokra gondolnunk. Az egy érdekes kérdés, hogy a hangvezérlés bekerül-e a gyártórendszerek üzemeltetésének napi gyakorlatába is. A cikkben a megbízhatóságot és az ismételhetőséget vizsgálták egy ember-robot

együttműködési folyamatban. A 2200 szimulációban lefuttatott analízis eredményét egy táblázatban foglalták össze, amely mutatja a kapcsolatot a sikeresen és a kevésbé sikeresen értelmezett parancsok között. A jól értelmezhető ideális parancs jellemzői: több mint egy szóból áll és nem túl rövid.

A digitális ikrek (digital twin) olyan szimulációs programok, amelyek online összeköttetésben vannak a fizikai gyártórendszerrel, annak minden egyes részével, és digitálisan lemásolják a fizikai rendszer működését. Ilyen digitális ikrek gyártásba integrálását vizsgálták Sobrino és társai, tanulmányukban [70]. Általánosságban a szimulációk egyik hiányossága, hogy a valós adatokkal nincsenek kapcsolatban, ezen kíván változtatni a szimuláció 4.0 a digitális ikrek koncepciót felhasználva. A technológia számos eszközt használ, IoT, big data, felhő szolgáltatások, és az aktuálisan elérhető adatok. A fejlesztett valós idejű lokációs rendszer (Real Time Location System - RTLS) növeli a rugalmasságot és a hatékonyságot a gyártás szintjén (shop floor) az adott időben.

Korábbi kutatásomban már készítettem egy szakértői keretrendszer prototípust, amely programmal hegesztő célgépek tervezését kívántam támogatni a már korábban tervezett és gyártott gépek alapján [71]. A rendszer egy mintát illesztett a felmerült vevői igényekre és ezeket vetette össze a tudásbázisban tárolt korábbi gépek mintáival. A pontos egyezéseket, eltéréseket és bizonytalanságokat egy FUZZY jellegű számhalmaz kezelte.

Simonova és társai gyártási technológiai útvonalak létrehozásához fejlesztettek egy szakértői rendszert, amely egy keretrendszert (frame model) használ [72]. Ennek a keretnek négy előnyét sorolták fel: az első, hogy a keret egy szabványos szituációt reprezentál, és a formális modellekkel szemben heurisztikus. A második, hogy a keretek szemantikai struktúráként az adott helyzetnek megfelelően hívhatók meg és illeszthetők. A harmadik, hogy a szemantikai hálóknak megfelelően öröklődnek a tulajdonságok, ami a hasonló technológiai folyamatok leírásához elég kényelmes. És végül a legfőbb előnye ennek a modellnek, hogy tükrözi az emberi emlékezet szerveződésének fogalmi alapjait, rugalmasságát és átláthatóságát.

1.14 Multi kritériumos kiválasztási módszerek

Vannak a robotok kiválasztásának már használt módszerei is. Ezek jellemzően a multi-kritériumos döntéstámogató módszerek (multi-criteria decision making, MCDM) közé tartoznak, mint pl.: a VIKOR módszer. Burcu és társai a VIKOR módszerrel oldják meg a robotok kiválasztási feladatát. A bemutatott fuzzy alapú módszer súlyozza az egyes tulajdonságokat, majd ezek alapján egy döntési mátrix segítségével rangsorolja az egyes robotokat, így a különféle tulajdonságok összevonva jelennek meg, és így előáll a robotok rangsora, legelején a követelményeknek leginkább megfelelővel [73].

Fu és társai tanulmányukban az MCDM módszereket terjesztették ki, többek között konszenzusos csoportos szakértői döntéssel, és annak sztohasztikus kiegészítésével [74]. Esettanulmányukban összevetették módszerüket VIKOR és ELECTRE II eredményekkel is, amivel igazolták, hogy fejlesztésük a hagyományos egyetlen összesített pontszámra alapuló módszerekkel szemben eltérő kiválasztási javaslatokat ad.

Egy szigetelő anyag kiválasztást vizsgáltak Ulutas is társai tanulmányukban, amelyben részletesen leírták a kiválasztási módszereket, amelyeket most röviden ismertetek [75]. A terület különbözik jelen kutatás tárgyától, a többkritériumos döntési példa viszont részletesen ismertetésre kerül, amiből a későbbiekben a kutatás során több elemet is felhasználtam. A három módszer, amit használtak:

- A PSI (Preference Selection Index) módszer objektív súlyokat használ a kritériumok értékelésére.
- CRITIC (gyógyszeripari vállalatok értékelésére lett fejlesztve [76]) módszer, amelyben a szórások mellett a kritériumok közötti korrelációt is figyelembe veszik.
- CoCoSo (Combined Compromise Solution) módszer célja, hogy kombinálja más MCDM módszerek erősségeit, és kiegyensúlyozottabb eredményt adjon azáltal, hogy kompromisszumokat kínál a különböző módszerek között.

A PSI módszer lépései:

- Normálás: a kritériumok típusától függően.

- Súlyozás: az átlagos értékek szórása alapján.

A döntési mátrix általános alakban:

$$D = [d_{ij}]_{m \times n} = \begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1n} \\ d_{21} & d_{22} & \cdots & d_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ d_{m1} & d_{m2} & \cdots & d_{mn} \end{bmatrix} \quad (22)$$

Meghatározzuk az egyes kritériumok haszon vagy költség jellegét, majd normalizáljuk a mátrix elemeit.

Haszon (pl.: fajhő) jellegű kritérium esetében:

$$d_{ij}^* = \frac{d_{ij}}{\max(d_{ij})} \quad (23)$$

Költség (pl.: sűrűség, hővezetés) jellegű kritérium esetében:

$$d_{ij}^* = \frac{\min(d_{ij})}{d_{ij}} \quad (24)$$

Ezután átlagolt normalizált értékeket számolunk (a „j”-edik kritériumra):

$$\bar{d}_j^* = \frac{\sum_{i=1}^m d_{ij}^*}{m} \quad (25)$$

Majd a preferencia variancia „PV_j” értékeket számoljuk ki:

$$PV_j = \sum_{i=1}^m (d_{ij}^* - \bar{d}_j^*)^2 \quad (26)$$

Következő lépésben az eltérést „δ_j” számoljuk ki:

$$\delta_j = (1 - PV_j) \quad (27)$$

A számított értékek alapján az adott kritérium súlya:

$$w_{jPSI} = \frac{\delta_j}{\sum_{j=1}^n \delta_j} \quad (28)$$

Összefoglalva elmondható a PSI módszerről, hogy a súlyozás az átlagos értékek és azok szórása alapján történik, a nagyobb eltérés kisebb súlyt eredményez.

A második módszer a CRITIC, amely az adott kritérium szórása mellett annak korrelációját is figyelembe veszi, ami úgy jelenik meg, hogyha két kritérium hasonló információt hordoz, vagyis együtt változnak, akkor az egyik súlyát lecsökkenti, így nem számítja be duplán azt az információt. A CRITIC módszer súlyképlete a (29) -es egyenlet szerinti.

$$k_j = \sigma_j * \sum_{e=1}^n (1 - r_{je}) \quad (29)$$

Ahol:

- σ_j : az adott kritérium szórása, vagyis mennyire változékony
- r_{je} : a „j-edik” és „e-edik” kritérium közötti korreláció
- $\sum(1 - r_{je})$: minél kisebb a korreláció, annál nagyobb a súlya

Amennyiben egy kritériumnak nagy szórása van és kevésbé korrelál másikkal, akkor nagy súlyt kap, amennyiben nagy a szórása, de erősen korrelál másikkal, akkor a súlya csökken. Ha viszont a szórása kicsi, akkor alacsony súlyt kap, mert a kicsi szórás a kevés információ tartalomra utal. Hasonló módszert használtak, jóval összetettebb jelöléssel Tarafdar és társai mobil robotok kiválasztására létrehozott keretrendszerükben [77]. Mind a TOPSIS mind a CoCoSo módszert fejlesztették, és ezek mérőszámait egyesítették egy RSI (Robot Selection Index – robot kiválasztási index) mutatóban. A cikk 2025 áprilisi megjelenése is mutatja a téma aktualitását, és elkülöníti a szubjektív és objektív tényezőket, amiket fuzzy technikával is kombinál, így jelentős előrelépést kínál az AMR-ek kiválasztásában a bizonytalanság kezelésében és a többkritériumos kiválasztás kritériumai közötti kompromisszumos kiegyensúlyozásában.

Kao és társai két csoportos döntéshozatali (kiválasztási) módszert vetettek össze tanulmányukban [78]. Az egyik egy ún. radiális modell, ahol a bemenetek generálják a kimeneteket. Matematikailag egyszerű maximum kiválasztásként írható fel, ami a (30) -ban szerepel.

$$\max uY_k; \text{ ahol } vX_k = 1; uY_j - vX_j \leq 0; j = A, B; u, v \geq \varepsilon \quad (30)$$

Az X-es tagok a bemenetek, az Y a kimenet, az u és v értékek a súlyozás, amiket egy ε küszöbértékkel (a tanulmányban nem-Archimédeszi számként említett) megegyező értéken, vagy a fölött kell tartani. Az egyes máshonnan nem ismert értékeket a humán szakértőkkel egyeztetve becsülik meg. A módszer problémája az ε küszöbértéktől való függése, ami miatt torzíthatja az eredményt, ha az értékek a hatékonysági határ közelében vannak. A másik módszer az SMB (Slacks-Based Measure) modell lényege, hogy nem arányosan skáláz, vagyis nem befolyásolja az ε küszöbérték. Az esettanulmány részében 18 robotot hasonlított össze három szakértő, és megállapításuk szerint az SMB módszer pontosabban tudja a szakértői véleményeket összegezni, így megbízhatóbb eredményt ad a kiválasztásra.

Raktárakban használt robotok kiválasztásáról írtak tanulmányt Bahadir és társai [73]. Ők a VIKOR (eredeti nyelven: ViseKriterijumska Optimizacija I Kompromisno Resenje, angolul: Multicriteria Optimization and Compromise Solution) módszert használták fuzzy kiegészítéssel. Ezt a módszert 2004-ben fejlesztették ki és publikálták [79]. Ennek a kompromisszumos döntési módszernek a célja, hogy az ideálishoz legközelebb lévő megoldást adja meg, konfliktusos kritériumok esetén. A módszer működésére felírok egy mintapéldát.

A példában 4 robotot hasonlítok össze 5 kritérium alapján: A1, A2, A3, A4 robotok. A kritériumok mellé oda írtam milyen jellegűek:

- C1 – Teherbírás (haszon)
- C2 – Sebesség (haszon)
- C3 – Ár (költség)
- C4 – Méret (költség)
- C5 – Energiatakarékosság (haszon)

Első lépésben létrehozom a döntési mátrixot, amihez normalizáljuk a kritériumokat jellegtől függően.

Haszon jellegű esetében:

$$f_{ij} = \frac{x_{ij} - x_j^-}{x_j^* - x_j^-} \quad (31)$$

Költség típus esetén pedig:

$$f_{ij} = \frac{x_j^* - x_{ij}}{x_j^* - x_j^-} \quad (32)$$

Ahol:

- x_j^* : a legjobb érték
- x_j^- : a legrosszabb érték

A súlyozásnál lehet az egyes kritériumok fontosságát egymáshoz képest beállítani, ez fontos lesz a későbbiekben. Egyelőre tekintsünk azonos súlyokat: $w_j = 0,2$, és a súlyok összegének 1-nek kell lennie.

Meghatározzuk az eltérések mértékét a (33) és (34) szerint.

$$S_i = \sum_{j=1}^n w_j * (f_j^* - f_{ij}): \text{ az ideálistól vett eltérés.} \quad (33)$$

$$R_i = \max_j [w_j * (f_j^* - f_{ij})]: \text{ a maximális eltérés.} \quad (34)$$

Végül az összesített Q_i értéket a (35) -ös egyenlet szerint tudjuk meghatározni.

$$Q_i = v * \frac{S_i - S^*}{S^- - S^*} + (1 - v) * \frac{R_i - R^*}{R^- - R^*} \quad (35)$$

Q_i : kombinált kompromisszumos mérőszám, ez alapján történik a rangsorolás, és „v” a stratégiától függő súlyozás, általános esetben $v=0,5$.

Egy konkrét példán is bemutatom a VIKOR módszer számolását, a számolásokat Excel táblázatban végeztem a korábban leírt (31) – (35) képleteket felhasználva. A 2. táblázat az adatokat mutatja, és a számolás első lépéseit is.

2. táblázat VIKOR módszer példa adatok, és számolás

Alternatíva	Kritériumok		
	Ár (eFt)	Sebesség (m/s)	Energiafogyasztás (kWh)
A1	100	1,2	2,0
A2	80	1,0	3,0
A3	90	1,4	2,5
A4	120	1,1	1,5
min.	80	1	1,5
max.	120	1,4	3
Alternatíva	f_{ij}		
A1	0,500	0,500	0,667
A2	1,000	0,000	0,000
A3	0,750	1,000	0,333
A4	0,000	0,250	1,000

S_i , és R_i számítása:

3. táblázat S_i és R_i értékek számítása

Alternatíva	S_i	R_i
A1	0,44444	0,1665
A2	0,66667	0,333
A3	0,30556	0,222
A4	0,58333	0,333

Q_i számításához a (35) képletben szereplő értékek:

4. táblázat Segédértékek Q_i számításához

v	0,5
S^*	0,306
S-	0,667
R^*	0,167
R-	0,333

A Q_i számítása az 5. táblázatban található.

5. táblázat A Q_i számított értékei

Alternatíva	S_i	R_i	Q_i
A1	0,44444	0,1665	0,190
A2	0,66667	0,333	1,000
A3	0,30556	0,222	0,165
A4	0,58333	0,333	0,884

Itt a legkisebb Q_i érték a legjobb választást jelenti, vagyis 0,165-tel az A3 robot került az első helyre a rangsorban, míg az A1-es robot a második helyre.

Bahadir és kutatótársai [73] a VIKOR módszert kiegészítették egy fuzzy minősítési rendszerrel is, amely azt mutatja meg, hogy az egyes paraméterek (teljes költség, sebesség, teherbírás, műveleti idő, funkcionalitás pl.: ember nélkül is fel tud venni termékeket magára a robot) mennyire fontosak (nagyon fontos, fontos, kevésbé fontos, stb.) az adott feladat szempontjából. Ennek értékeit a szakértők adták meg, majd a módszer fuzzy része alakította át ezeket a tulajdonsággal megadott értékeket számértékké, vagyis súlyokká. Itt kapott szerepet az emberi szakértelem, ami szubjektivitást visz a módszertanba. Ezt a szubjektivitást kompenzáló három szakértő véleményét átlagolták. Ez a fuzzy alapú módszer súlyozza az egyes tulajdonságokat, majd ezek alapján egy döntési mátrix segítségével rangsorolja az egyes robotokat, így a különféle tulajdonságok összevonva jelennek meg, és így előáll a robotok rangsora, lelegején az ideállal.

Chodha és társai egy ipari ívhegesztőrobot kiválasztását vizsgálták MCDM technikával [80]. A cikk a TOPSIS (Technique for Order Preference by Similarity to Ideal Solution – az ideális megoldáshoz való hasonlóság szerinti sorrendpreferencia technikája) módszert használta, amely az ideális és a legkevésbé ideális végletek közötti távolságok alapján rangsorolja az alternatívákat. A kritériumok súlyozásához entrópia módszert használ, amely az adott információk alapján objektíven határozzák meg a súlyokat. Nyolc robotot hasonlítottak össze, tíz szempont szerint (pl.: költség, sebesség, terhelhetőség és pontosság). A költségeknél a minél kisebb, míg a pontosságnál és a

teljesítménynél a minél nagyobb értékek voltak jobbak, ami analógiája a korábban tárgyalt költség és haszon jellegű kritériumoknak.

A TOPSIS módszer lényege és lépései röviden:

- A döntési mátrix kialakítása: a kritériumokat (pl.: ár, pontosság, terhelhetőség) az alternatíváknak (vagyis a különféle robotmodelleknek) megfelelően egy mátrixba rendezik.
- Az értékeket egységes skálára hozzák (pl.: 0-1), hogy összehasonlíthatóak legyenek.
- Minden kritériumhoz egy súlyt rendelnek, amely kifejezi annak fontosságát, objektív (pl.: entrópia módszerrel) vagy szubjektív (szakértői értékelés alapján) módon.
- Az ideális megoldás: minden kritériumból a legjobb érték (pl.: legkisebb ár, legnagyobb pontosság)
- Legkevésbé ideális megoldás: minden kritériumból a legrosszabb érték
- Kiszámítják az ideálistól és a legkevésbé ideálistól való távolságot az egyes alternatívákra.
- Kiszámítják a relatív közelséget, a fenti kettő felhasználásával.
- Végül rangsorolják az egyes alternatívákat a relatív közelség alapján.

6. táblázat TOPSIS módszerhez paraméterek

Robot	Ár (millió Ft)	Sebesség (m/s)	Terhelhetőség (kg)	Pontosság (mm)	Energiafogyasztás (kW)
A1	15	1,5	10	0,05	5,2
A2	12,5	1,2	12	0,04	4,8
A3	18	1,6	11	0,03	5
A4	14	1,3	9	0,06	5,5
Jobb	kisebb	nagyobb	nagyobb	kisebb	kisebb

A módszer működését egy kisebb példával is szemléltem, hogy jobban érthető legyen, és később is fel tudjam használni. Adott 4 robot, és adottak a szempontok, a 6. táblázat szerinti értékek objektív értékekkel.

Az egyes kritériumok súlyai:

7. táblázat Kritériumok súlyozása

Ár (millió Ft)	Sebesség (m/s)	Terhelhetőség (kg)	Pontosság (mm)	Energiafogyasztás (kW)
0,25	0,2	0,2	0,2	0,15

A normalizálás folyamata minden egyes kritériumra:

$$r_{ij} = \frac{x_{ij}}{\sqrt{\sum x_{ij}^2}} \quad (36)$$

Az árra a (37) szerinti összefüggést használjuk.

Legkisebb ár: 12,5 millió Ft, ezt írom a számlálóba amikor a „min” a jobb eset (korábban költség és hasznon néven szerepelt, a min a költség jellegre utal).

$$r_{ij} = \frac{12,5}{x_{ij}} \quad (37)$$

Az összes „min” jobb esetben így alakulnak a normalizált értékek:

8. táblázat Normalizált értékek a „min” jobb esetben

Robot	Ár (millió Ft)	Ár normalizált értéke	Pontosság (mm)	Pontosság normalizált értéke	Energia-fogyasztás (kW)	Energia-fogyasztás normalizált értéke
A1	15	0,833333333	0,05	0,6	5,2	0,923076923
A2	12,5	1	0,04	0,75	4,8	1
A3	18	0,694444444	0,03	1	5	0,96
A4	14	0,892857143	0,06	0,5	5,5	0,872727273

A „max” jobb esetben (korábban hasznon jellegű) használandó összefüggés, pl.: sebesség esetében:

$$r_{ij} = \frac{x_{ij}}{1,6} \quad (38)$$

9. táblázat Normalizált értékek a „max” jobb esetben

Robot	Sebesség (m/s)	Sebesség normalizált értéke	Terhelhetőség (kg)	Terhelhetőség normalizált értéke
A1	1,5	0,9375	10	0,833333333
A2	1,2	0,75	12	1
A3	1,6	1	11	0,916666667
A4	1,3	0,8125	9	0,75

Ezek után minden normalizált értéket megszoroztam a hozzá tartozó súllyal, pl.: az A1 normalizált árát az árhoz tartozó súllyal a (39) szerint, majd a kapott értékeket beírtam a 10. táblázatba.

$$0,833 * 0,25 = 0,208 \quad (39)$$

10. táblázat Normalizált és súlyozott értékek

Robot	Ár normalizált értéke	Ár normalizált és súlyozott értéke	Sebesség normalizált értéke	Sebesség normalizált és súlyozott értéke	Terhelhetőség normalizált értéke	Terhelhetőség normalizált és súlyozott értéke	Pontosság normalizált értéke	Pontosság normalizált és súlyozott értéke	Energia-fogyasztás normalizált értéke	Energia-fogyasztás normalizált és súlyozott értéke
A1	0,8333	0,2083	0,9375	0,1875	0,8333	0,1667	0,6	0,12	0,9231	0,1385
A2	1	0,25	0,75	0,15	1	0,2	0,75	0,15	1	0,15
A3	0,6944	0,1736	1	0,2	0,9167	0,1833	1	0,2	0,96	0,144
A4	0,8929	0,2232	0,8125	0,1625	0,75	0,15	0,5	0,1	0,8727	0,1309

Az ideális „A+” és legkevésbé ideális „A-” megoldások a 11. táblázatban szerepelnek.

11. táblázat Szélsőértékű megoldások

Kritérium	A+ (legjobb)	A- (legrosszabb)
Ár	1,00	0,694
Sebesség	1,00	0,75
Terhelhetőség	1,00	0,75
Pontosság	1,00	0,5
Energia fogyasztás	1,00	0,873

Euklideszi távolság számítása (négyzetösszeg) az ideálistól:

$$D_i^+ = \sqrt{\sum_{j=1}^n (v_{ij} - v_j^+)^2} \quad (40)$$

ahol:

- v_{ij} : az adott robot súlyozott normalizált értéke az „i”. robotnál, és a „j”. kritériumnál.

- v_j^+ : az ideális (legjobb) súlyozott érték az adott kritériumban.

A súlyozott szélső értékek szerepelnek a 12. táblázatban.

12. táblázat Súlyozott szélsőértékek

Kritérium	Súlyozás	A+ (legjobb)	Súlyozott A+	A- (legrosszabb)	Súlyozott A-
Ár	0,25	1,00	0,25	0,694	0,1735
Sebesség	0,2	1,00	0,20	0,75	0,15
Terhelhetőség	0,2	1,00	0,20	0,75	0,15
Pontosság	0,2	1,00	0,20	0,5	0,1
Energia fogyasztás	0,15	1,00	0,15	0,873	0,13095

A „D⁻” számításakor az „A⁻” -t vontam ki, így kaptam meg a 13. táblázatot.

13. táblázat A D⁺ és D⁻ értékei

Robot	D ⁺ (ideálistól mért Euklideszi távolság)	D ⁻ (a legkevésbé ideálistól mért Euklideszi távolság)
A1	0,09765556	0,057912107
A2	0,070710678	0,105901617
A3	0,078415816	0,117394318
A4	0,122426252	0,0512617

Utolsó lépésben a relatív közelséget számoltam ki, ez „C^{*}” -al van jelölve:

$$C^* = \frac{D^-}{(D^+ + D^-)} \quad (41)$$

14. táblázat A C^{*} számítása

Robot	D ⁺ (ideálistól mért Euklideszi távolság)	D ⁻ (a legkevésbé ideálistól mért Euklideszi távolság)	C [*] (relatív közelség)
A1	0,09765556	0,057912107	0,372263
A2	0,070710678	0,105901617	0,599628
A3	0,078415816	0,117394318	0,599531
A4	0,122426252	0,0512617	0,295137

Végül az eredmények a 14. táblázatban láthatóak. Ebből a táblázatból az látszik, hogy az „A2” robot 0,5996-os értéke a legmagasabb. Az „A3”-as robot 0,5995-ös értéke szinte ezzel megegyező, vagyis ez a két robot a legmegfelelőbb a feladatra. Mivel az „A3”-as robot ára 18 millió Ft az „A2” -es roboté viszont csak 12,5 millió Ft ezért az „A2”-es robot kiválasztása javasolható. Az „A3” -as robot előkelő helyezése a kiugróan magas ára miatt nem indokolt (mivel ez költségként szerepel, vagyis a „min” jobb kategóriában van), ugyanakkor ez a robot a legpontosabb és a leggyorsabb is, ezért kapott ilyen magas pontszámokat.

Érdekesség képen megnéztem, hogy az „ár” kritérium súlyának 0,4 -re emelésével hogyan változik a sorrend. Ekkor a sebesség 0,1 az energia fogyasztás szintén 0,1 -es súlyt kapott, a pontosság és a terhelhetőség továbbra is 0,2 súllyal szerepeltek. A 15. táblázatban látható eredményeket kaptam.

15. táblázat Alternatíva eredmények

Robot	D+ (ideálistól mért Euklideszi távolság)	D- (a legkevésbé ideálistól mért Euklideszi távolság)	C* (relatív közelség)
A1	0,109789752	0,064502881	0,370084
A2	0,055901699	0,141926213	0,717423
A3	0,123418189	0,108682256	0,468255
A4	0,121861728	0,079788028	0,395676

Vagyis az „A2” -es (6. táblázatban szereplő legolcsóbb) robot kiemelkedik a mezőnyből, ahogyan arra számítani is lehetett.

Chodha és társai cikkükben [80] egy entrópia alapú megközelítést is alkalmaznak, ami röviden arra utal, hogy amelyik értéknek nagyok az eltérései az átlagtól, vagyis nagy a szórása, az nagyobb információval bír ezért nagyobb súlyt kap. Amennyiben az adott kritérium értékei nagyon hasonlóak az összes alternatívára, akkor az kevés információval bír, így kisebb súlyt kap. Ez megfelel annak a szemléletnek, hogy az entrópia, egyszerűen megfogalmazva „a rendezetlenség mértéke”. Ugyanakkor így teljesen objektív az értékelés, a szakértői súlyozás nem jelenik meg. Természetesen ez azt is jelenti, hogy egy adott gyártórendszerre szabott feltételeket sem tudjuk figyelembe venni, ez csak a kritériumok súlyozásánál jelenik meg.

Goswami és társai analíziseket végeztek újonnan kifejlesztett hibrid MCDM modelleken [81]. Módszerükben az alábbiakat kombinálják:

- DEMATEL (Decision Making Trial and Evaluation Laboratory), amely célja szétválasztani a kritériumokat ok és okozati típusú csoportokra.
- ANP (Analytic Network Process), amely a kritériumok közötti kölcsönös függőségre világít rá.
- TOPSIS módszert használnak a végső rangsor felállítására.

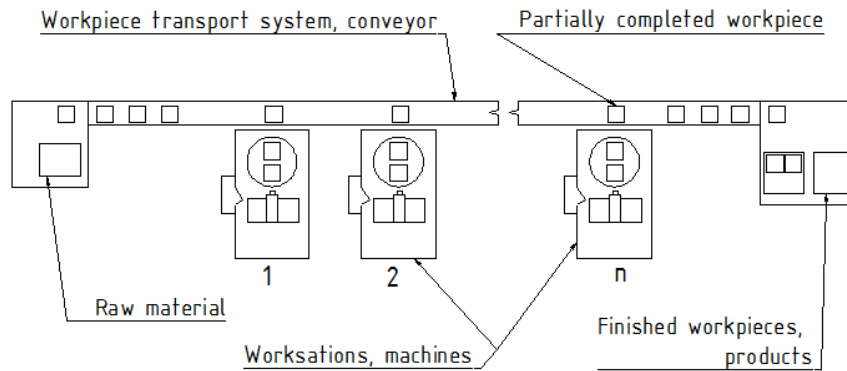
2 A Mobil robot központú modell fejlesztése

A kutatásom során készítettem egy saját gyártórendszer modellt. A modellben egy autó indítókulcs termék összeszerelését vizsgáltam azért, mert ez sok szempontból prezentál egy általános szerelési folyamatot. Volt benne alapanyag kezelés, szállítás, összepattintás, gravírozás, csavarozás, lézeres jelölés, 3D nyomtatás és csomagolás is. A minta gyártórendszerből az egyik leszűrt tanulság, hogy a mobil robot számára a folyamat leegyszerűsíthető két pont közötti szállítási feladat megfogalmazására. A saját gyártórendszer modellem lényege, hogy csak a mobil robot szempontjából releváns információkat tartalmazza, vagyis az egész rendszerre a mobil robot szemszögéből néz rá. A gyártórendszerek modellezésénél említésre került, hogy az egyik modellezési lehetőség a diszkrét eseményű rendszerek alapján felírt modellek használata. A kutatás részeként elvégeztem a diszkrét eseményű rendszerek modellezési megoldásainak feltérképezését is. Az egyik módszer a véges állapotú automaták, a másik pedig a Petri-hálók.

A továbbiakban termelési rendszer alatt megmunkáló vagy összeszerelő berendezések sorát értem, amelynek célja egy adott termék (kész vagy félkész) előállítása, adott erőforrások (pénz, idő, nyersanyag, szakértelem) felhasználásával. A saját modell felírásának magyarázatához a fejlődésük sorrendjében mutatom be a gyártórendszereket.

2.1 Anyagáramlás megvalósítása szállítószalaggal

A harmadik ipari forradalmat követő automatizálási trendhez kapcsolódó gyártási háttér az jellemezte, hogy a nagyüzemi és tömeggyártási rendszerek során vagy teljesen azonos, vagy nagyon hasonló (pl.: típuscsaládok) alkatrészeket és termékeket állítottak elő. A lineárisan tervezett gyártórendszerek kiváló megoldást jelentettek ennek a követelmények a kielégítésére, mivel a gyártási művelet sorrendje rögzített volt. Az ilyen célokra tervezett gyártórendszerek tipikus munkadarab-átadási megoldása a szállítószalag volt, és a félkész terméket a szállítószalagról emberi erővel vagy robot segítségével juttatták át az adott cellába. Ez a gyártórendszer felépítés látható a 6. ábrán.

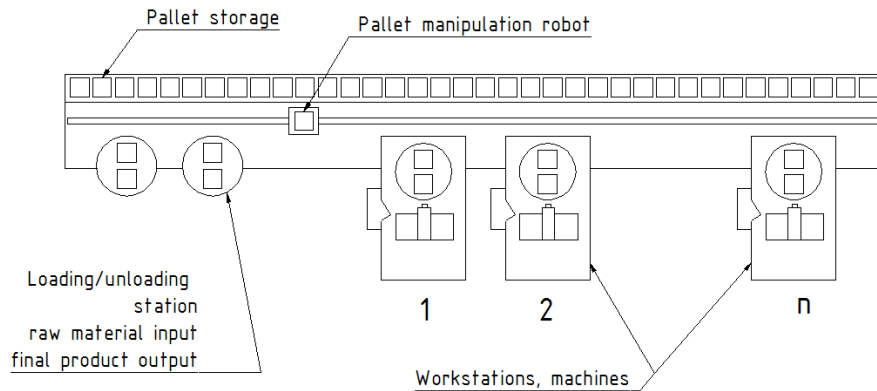


6. ábra Hagyományos szállítószalagos gyártórendszer felépítése

Általánosan felírható véges állapotú automataként egy gyártórendszer. Ebben az esetben egyes állapotokból léphetünk másik állapotokba adott bemenő jelek hatására. A teljesebb kifejezőmód és a további kiterjesztési lehetőségek miatt én a Petri-hálós modellt választottam. A legegyszerűbb esetre vonatkozó Petri-hálós modell szerint egyetlen elágazás sincs, hiszen az úgynevezett konzervatív struktúrákban kiépülő gyártórendszerek esetén egyáltalán nem generálódott igény az anyagáramlás rugalmassá tételére.

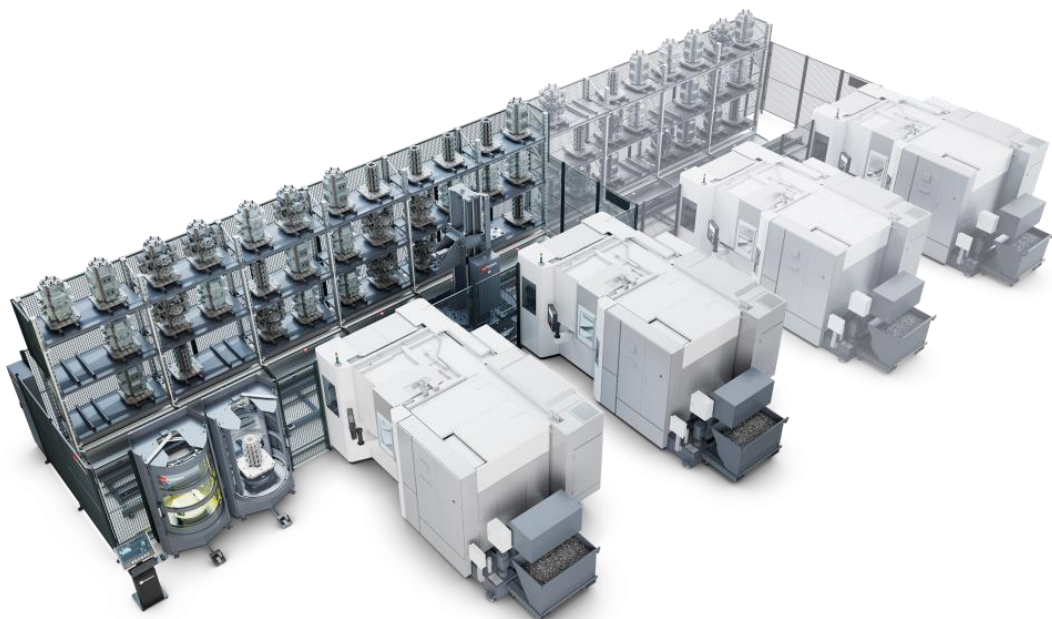
2.2 Flexibilis gyártórendszer

A következő minőségi ugrás a gyártórendszerek fejlődésében az volt, amikor a munkadarabok nem lineárisan helyezkedtek el a szerszámgépekhez hasonlóan, hanem a rendszer egy pufferraktárból választotta ki a megmunkálendő darabokat, és egy központi manipulátor juttatta el azokat a megmunkálógépekhez. Ezt nevezzük rugalmas gyártórendszernek, vagy röviden FMS-nek, ami a Flexible Manufacturing System kezdőbetűiből jön. Ez növelte a gyártás rugalmasságát, egyes műveleteket párhuzamossá tett, míg más műveleteket el lehetett hagyni. Ez a gyártási rendszerstruktúra már sokkal közelebb áll a negyedik ipari forradalom igényeihez, amely ezen részterületen azon alapul, hogy az integrált gyártási rendszerek a lehető legkülönbözőbb geometriákat, méreteket és formákat képesek előállítani, sok esetben időben átfedve egymást. Ennek a rugalmas gyártási rendszernek a vázlatát a 7. ábrán látható.



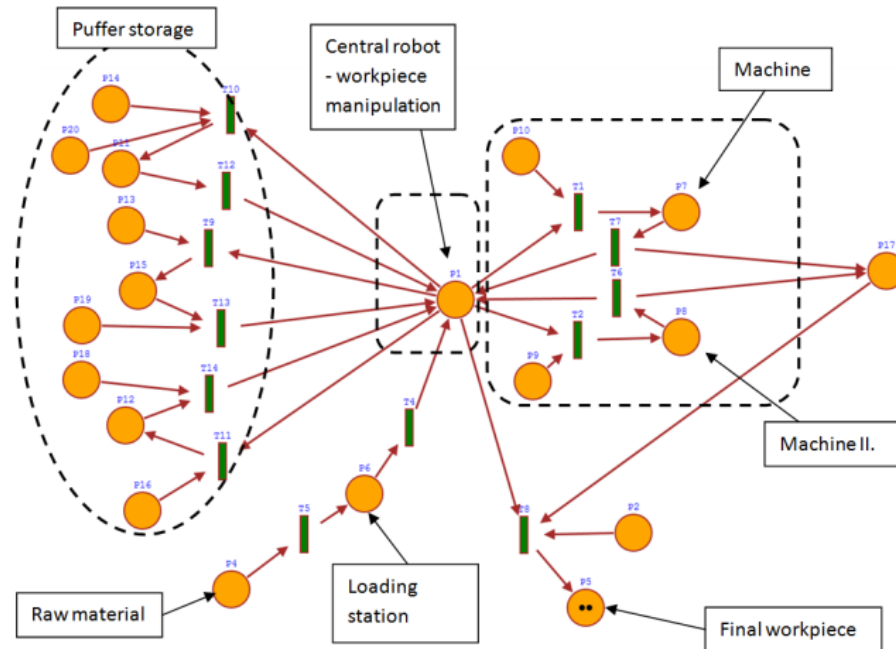
7. ábra Rugalmas gyártórendszer sematikus rajza

A munkadarabok mozgását egy lineáris pályán mozgó robot végzi. Minden munkadarab egy palettán van elhelyezve, így pontosan pozícionálható a szerszámgépben és a tároló rendszerben is. Ezeknek a palettáknak szabványos mérete van, egy jellemző méret pl.: 630mm x 630mm. A tárolóban több szinten is el lehet helyezni az alkatrészeket, így a helykihasználást maximalizálni lehet. Ennek a központi robot mérete szab csak határt. A könnyebb érthetőség kedvéért egy valós rendszert láthatunk a 8. ábrán. (kép forrása: <https://www.fastems.com/revolutionizing-4-and-5-axis-fps/> 2025.05. 20.) A képen látható még az is, hogy a teljes gyártórendszer könnyen bővíthető további gépekkel, ezt szimbolizálja a két jobb oldali gép (és ahhoz tartozó paletta tároló) halványabb képe.



8. ábra Valós FMS rendszer képe

Az említett modellezési eszközökből kiválasztottam a Petri-hálós modellezési módszert, és ennek segítségével felrajoltam az FMS rendszer lényegét megragadó Petri-hálót. Ez látható a 9. ábrán.

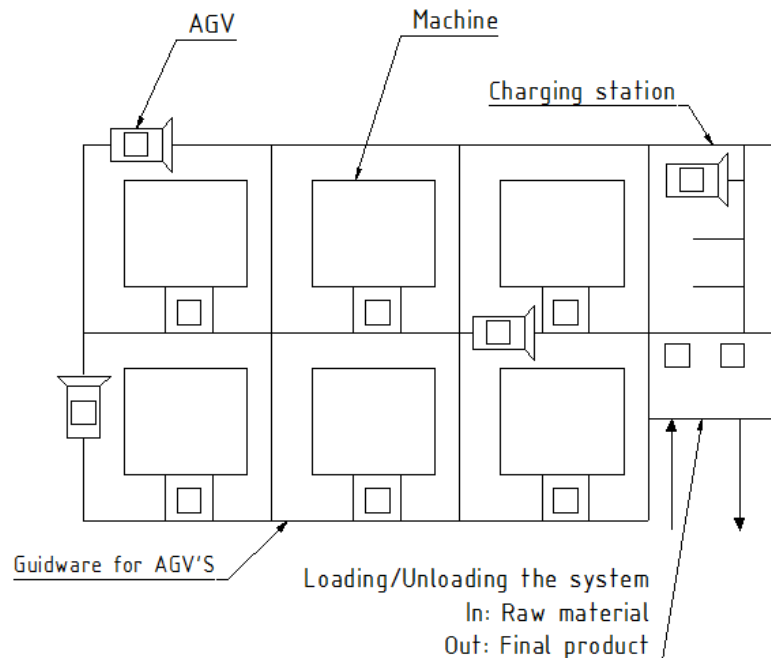


9. ábra FMS gyártórendszer Petri-hálója

A modell alapján készült gyártórendszerek esetében a rendszer központi eleme a paletta szállító manipulátor. A rendszert palettákkal a rakodóállomáson lehet feltölteni a munkadarabot hordozó palettákkal. Ezt követően a robot a palettát a tárolóba helyezi, az egyik polcra. Működés közben a palettákat a robot kiveszi a tárolóból, és az ütemezésnek megfelelő szerszámgéphez viszi. A forgácsolási folyamat befejezése után a munkadarabot a palettával együtt a manipulátor kiveszi a gépből, és visszahelyezi a tárolóba. A folyamat utolsó lépéseként a kész darabokat a kirakodó állomásra szállítja a manipulátor, ahonnan elszállíthatók. A gyártás megkezdésének bemeneti feltételei: a nyersanyagok palettán a tárolóban kell lenniük, a gépet fel kell szerelni a szükséges szerszámokkal, és a megfelelő programnak rendelkezésre kell állnia a gépen. Ha az egyik szerszámgép meghibásodik, egy másik szerszámgép átveheti a helyét, ha azon a gépen a fenti feltételek teljesülnek. Ez a funkció növeli a rendszer rugalmasságát és megbízhatóságát.

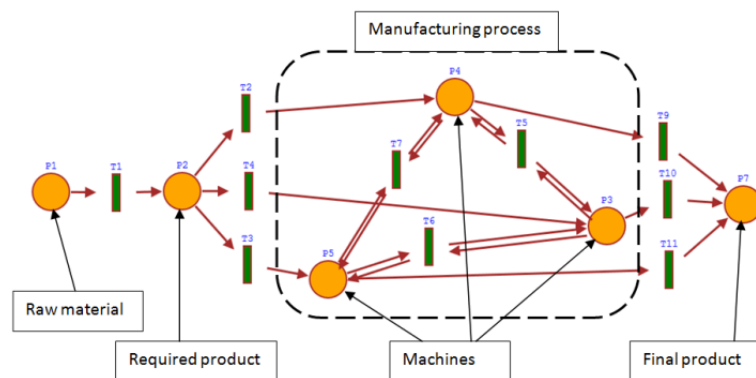
2.3 Automatikus vezérlésű járművel kiegészített gyártórendszer

A flexibilis gyártási rendszerek egyik fő korlátja maga a paletta kezelő központi manipulátor vagy robot, amely meghatározza a kiszolgálás sebességét és az egész rendszer optimális (maximális) méretét, figyelembe véve a manipulációt és a ciklusidőt. Ez növelhető több robot rendszerbe helyezésével. A maximális méretkorlátra megoldást jelenthet a szigetszerű termelési rendszer kialakítása, ahol mobil robotok (újabbán Automated Mobile Robot – AMR, korábban Automated Guided Vehicle – AGV) szállítják a raklapokat a termelési szigetek és a gyártócellák között. A 10. ábrán az AGV-k által kiszolgált gyártórendszer általános vázlatja látható.



10. ábra AGV-vel kiegészített gyártórendszer

A rendszer modellje a 11. ábrán látható. Ez a modell a gyártási folyamatok és a munkadarabok kapcsolatát mutatja csak be, maga a munkadarab továbbítása, vagyis az AGV vagy az AMR nem szerepel benne.

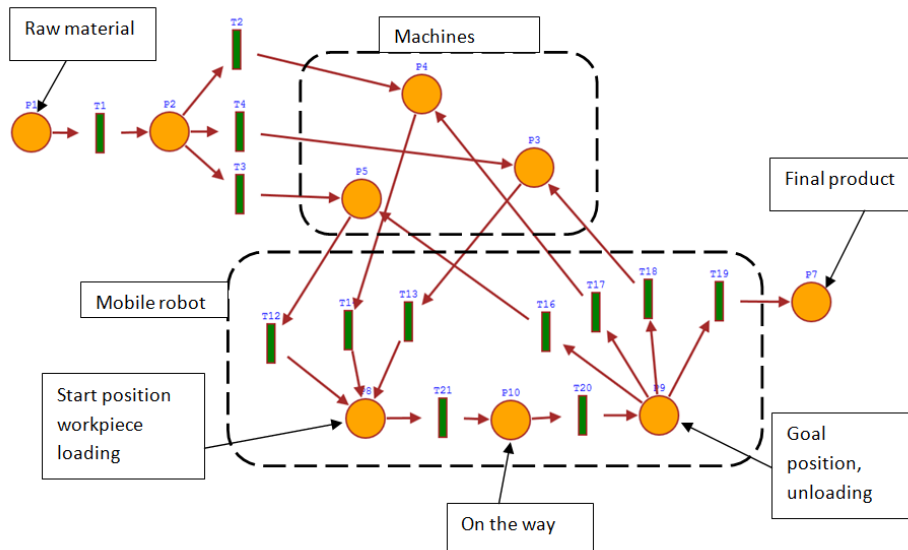


11. ábra Automatikus munkadarab mozgás modellje

Jelen kutatás egyik célja a mobil robotokat központba állítva, kialakítani egy új típusú gyártórendszer modellt. Ehhez első lépésben szét kell választani a közös modellben magát a gyártórendszert az azt kiszolgáló robotokat. A szétválasztási, majd ismételt összekapcsolási műveletek eredményeképpen összeáll a 12. ábrán látható, új típusú megközelítés, ahol a modellben bármely gépek között tetszőleges sorrendben mozoghat a munkadarab.

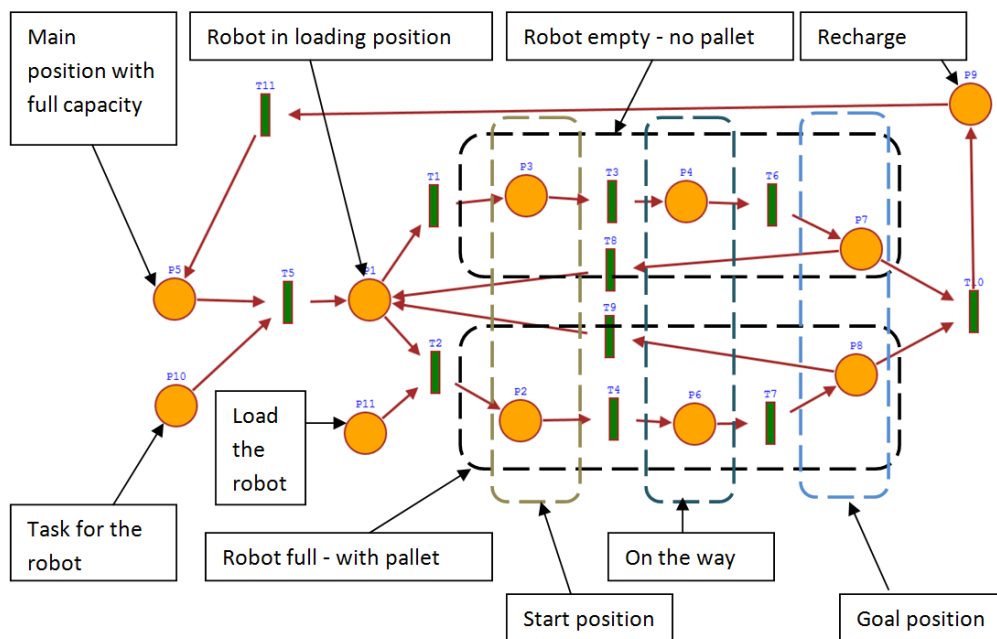
2.4 Robottal kiegészített modell

Az előző lépésben a robotokat leválasztottam a modelltől, így a gépek „csak”, mint a robotok által mozgatott termékek kibocsátói vagy fogadói jelennek meg. Majd bevezettem a munkadarab mozgását robotokkal, és berajzoltam a robotot a modellbe. Ezután szétválasztottam a gépeket és robotot. A 12. ábrán látható koncepció szerint a robot folyamatai három szakaszra különíthetők: a kiindulási pozíció, a robot úton van, és a cél pozíció. Mindhárom gép lehet kiindulási és végpozíció is. A termék elkészülte is az egyik végpozíció. A modell egyszerűsítve tartalmazza a folyamat elemeket, itt a lényeg a robot folyamatainak leválasztása az egész gyártórendszer modelltől.



12. ábra Gyártórendszer modell mobil robottal

Ezt követően felrajzoltam a robot folyamatait részletesebben, megkülönböztetve a start és cél pozíciókon kívül a robot üres és terhelt állapotát.



13. ábra A mobil robot modellje a gyártórendszerhez kapcsolva

A 13. ábrán látható ez a fejlesztett modell, ami mutatja a robot szempontjából a gépkiszolgálás folyamatát. A robot az egyik feladóhelytől megy a másikig, munkadarabbal vagy anélkül, ezt jelölik a szaggatott vonalak. Amikor utasítást kap a robot akkor odamegy a feladóhelyre (ez lehet egy kiindulási vagy várakozási pozíció is) majd a feladattól függően felveszi a terméket és elindul a

célállomás felé. A cél pozícióban lerakja a terméket, majd a modell szerint ekkor a töltőhelyre (vagy egy köztes helyre) megy, és utána visszalép a kezdeti pozícióba a modell, vagyis a korábbi célállapota lesz az új kiindulási állapot, és a folyamat kezdődik előlről. Ezt a modellt ismertettem a [82] publikációmban.

Különbéle mobil robotok különféle feladatokra alkalmasak. A leggyakoribb feladat a munkadarab mozgatása a raktár és az egyes állomások között. A Petri-hálók jellegzetessége, hogy az alapmodell kiterjeszhető. A leggyakoribb kiterjesztés az időzített Petri-háló, de létezik színezett és korlátozásokat kezelő változat is. Jelen kutatás szempontjából ez azt jelenti, hogy az egyes állapotokhoz lehet feltételeket is rendelni. Ilyen felétel lehet pl. a munkadarab tömege és mérete. A robotra vonatkozó paraméterek, amelyeket a kibővített modell tartalmazhat: a robot terhelhetősége, sebessége, akku kapacitása, de lehet a gyártó, a bekerülés és üzemeltetési költség is. Továbbá lehet értelmezni a különféle robotok kompatibilitását az egyes gépekkel is mind hardver mind szoftver oldalról. Egyes robotok navigációja nem igényel környezeti változtatásokat a telepítéskor. Másoknak szükséges lehet pl. felragasztani különféle markereket akár a padlóra, falra esetleg a mennyezetre. Ezeket az integritás témakörében lehet a kiterjesztett modellbe beépíteni.

A modell általánossága miatt többféle forgatókönyvet is támogat, sőt előre nem tervezett forgatókönyvek is kikövetkeztethetők a modellből, melyekre a konkrét feladat megoldásakor már fel lehet készülni.

A gyártórendszerben a mobil robot szempontjából fontos paraméterek:

- honnan és hova kell szállítani,
- mit kell szállítani: fizikai méretek, súly, darabszám,
- milyen útvonalon,
- mekkora távolságra,
- hogyan tudja felvenni és lerakni a robot,
- mekkora az akkumulátor töltés csökkenése a szállítási feladat hatására.

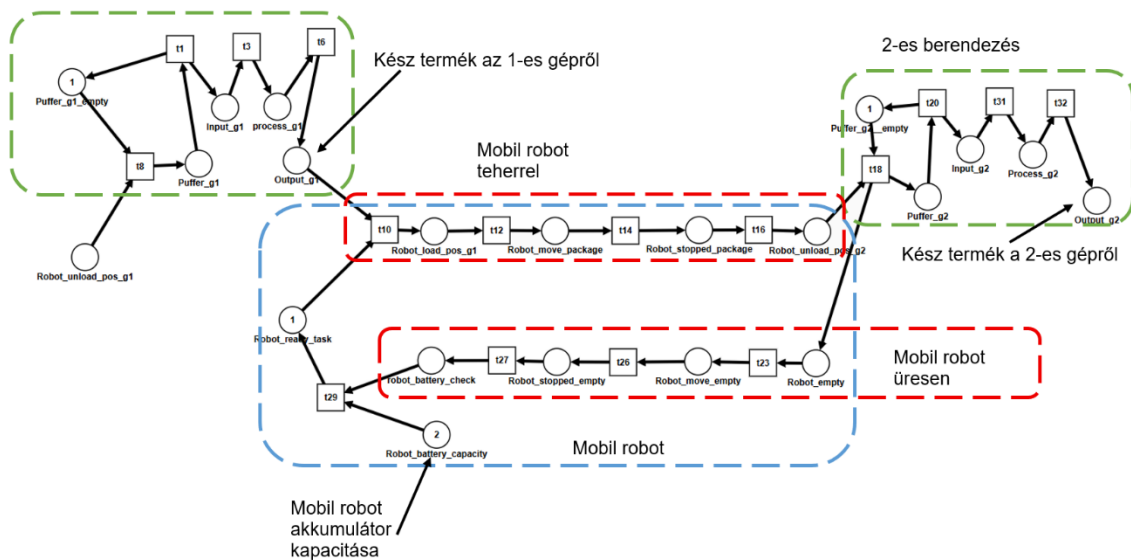
Amit ez a modell nem kezel, emiatt elhanyagolásra kerül:

- több robot esetén a robotok esetleges találkozása az útvonalon,
- szükséges robotok számának meghatározása,

- teljes gyártórendszer jellemzők vizsgálata.

A "feladóhely" lehet a gyártórendszer egy konkrét gépe, gyártósziget, szállítószalag, selejt tárló, mérőlabor, vagy akár puffertár is. A robot szempontjából nézve a feladatok végrehajtásán van a hangsúly, ezek mérőszámait lehet a modellel meghatározni és majd a konkrét esetben mérni. Ilyen mérőszámok lehetnek: a roboton mennyi ideig van munkadarab, mennyi időt tölt szállítással, várakozással, töltéssel, mennyire van kihasználva a robot kapacitása.

A Petri-hálós modell alkalmas a holtponatok felderítésére is.



14. ábra Mobil robot és gyártórendszer kapcsolat Petri-hálós alap modellje

A 14. ábrán látható Petri-hálós modellt tekintem az alapmodellnek a mobil robot központú gyártórendszer modellhez. Ezen két gép közötti szállítás megvalósítása látható. A két berendezés zölddel van keretezve, és az egyszerűsített belső működés is mindkét gép esetén Petri-hálóval fel van tüntetve. A mobil robot szempontjából ez két pontot jelent, az egyikről felveszi, a másikra lerakja a darabot. Közben megvalósul a szállítási feladat.

A robot modellben a következő állapotok vannak (körrel jelölve):

- Robot töltési pozícióban az egyes gépről (G1)
- Robot csomaggal úton van
- Robot megáll a csomaggal
- Robot lerakási pozícióban a kettes gépnél (G2)

- Robot üres
- Robot üresen mozog
- Robot üresen megáll
- Robot akkumulátor töltést ellenőriz
- Robot akkumulátor töltése
- Robot feladat teljesítésre kész

A robot modellben a következő átmenetek vannak (négyzettel jelölve):

- Robot feladat felvételi pozícióban várakozva megkapja a feladatot (t10)
- Robot elindul a csomaggal (t12)
- Robot megállásra kész a csomaggal (t14)
- Robot csomag leadásra készen (t16)
- Robot üresen elindul (t23)
- Robot üresen megáll (t26)
- Robot akkumulátor ellenőrzésre készen (t27)
- Robot akkumulátor kapacitás csökkentése (t29)

A szimulációs eszközzel az egyes átmenetekre kattintva lehet azok tüzelését aktiválni, amennyiben a megelőző helyeken megfelelő számú token rendelkezésre áll. Így pl.: az akkumulátor kapacitás nevű állapoton szükséges legalább egy tokennek lennie, hogy a robot a következő (feladatra kész) állapotába léphessen.

Petri Net Analysis				
Perform various tests on a petri net at once.				
bcf Yes	bicf Yes	output_nonbranching Yes	isolated_elements No	nonpure_only_simple_side_conditions No
pure Yes	strongly_connected No	simply_live No	backwards_persistent Yes	t_net No
reversible Yes	k-bounded 2	persistent Yes	t_net Yes	free_choice Yes
conflict_free Yes	k-marking 1	restricted_free_choice No	bounded Yes	homogeneous Yes
strongly_live No	safe No	asymmetric_choice Yes	weakly_live No	weakly_connected Yes

CLOSE START TESTS

15. ábra Alap modell Petri-háló analízis eredmények

A háló analízisének eredményét a 15. ábrán láthatjuk. Ezek alapján az alapmodell korlátos, konfliktus mentes, nem biztonságos és nincsenek benne elérhetetlen állapotok.

Az alapmodell kiterjesztésére az alábbi megfontolásokat tettem:

- Az éleknek lehet súlyt, és szín korlátot is adni:
 - A súly jelentheti a terhelés minimális számát, ha egyszerre több rakomány szállítására is használni akarjuk a robotot.
 - Lehet az útvonalakat priorizálni kisebb súllyal.
 - A mozgathoz szükséges akkumulátor kapacitás szintén lehet az élek súlyával arányos. Vagyis messzebbre történő szállításhoz lehet több tokenre is szükség.
- Színezett Petri-hálóval kiterjeszthető a modell az alábbi funkciókkal:
 - Az egyes színek jelenthetik az adott teher fizikai méreteit vagy tömegét.
 - Színekkel lehet a felvételi és leadási pontokhoz is funkciót rendelni, vagyis, pl.: hogy a robotnak képesnek kell-e lennie a földről felvenni vagy a földre lerakni a rakományt.
 - Hasonló módon lehet kezelni a munkamagasságot is.
- Időzített Petri-hálóval a modellt kiterjeszthetjük teljesítmény mérésére.
 - A szállításhoz, a mozgáshoz és a töltéshez szükséges időket mérhetővé tehetjük, így az egyes rakományok szállításához szükséges idők összevethetővé válnak, így a korábbi fejezetben megismert teljesítmény mutatók értelmezhetőek.

2.5 Gyártórendszer és mobil robot modellek kapcsolódási pontjai

A vizsgálatok fókuszja a mobil robot szerepe a gyártási folyamatban. Ehhez a fenti robotmodellt a gyártórendszerhez kell kapcsolni. Ez azért jobb, mint az eddigi megközelítések, mert így egy gyártórendszerhez többféle robot is gyorsan hozzákapcsolható és ezek egymással összehasonlíthatóak. A katalógusban megadott adatokon túl értékelni és mérni lehet a különféle robotok kihasználtságát, megbízhatóságát, vagyis, hogy melyik robot mennyire vált be az adott feladatra. A költségeket elemezve választ kaphatunk arra is, hogy érdemes-e beruházni az adott feladatra egy mobil robotra, vagyis megtérül-e adott idő alatt a befektetés (ROI – Return on Investment). A döntéskor nem korlátozódunk a

meglévő struktúrára, hanem kiterjesztjük a lehetőségeket, hiszen a modellünk számos változatot gyorsan tud generálni és összehasonlítani, akár olyanokat is, amikhez a gyártási folyamatokat, gépeket, alapanyagáramlást, át kell szervezni a meglévőhöz képest. Ez a modell lehetővé teszi a robotok különféle feladatainak lépésenként történő végig járását. Azonos feladatok mellett olyan feladatokra is van lehetőség, hogy pl.: egy robot miután felvett egy munkadarabot az egyik állomáson, utána még felvehet egy másik darabot egy másik állomáson, és a két termékkel „megpakolva” megy a leadási pozícióba.

Kutatásom célja, egy egyszerű, de kifejező modell felírása, mellyel vizsgálható a gyártórendszereket kiszolgáló mobil robotok sokasága. Először a gyártórendszerek fejlődésén keresztül vizsgáltam azok felépítését, majd leválasztottam róla a kiszolgáló ipari mobil robot modelljét.

A modellek kapcsolódása lehetővé teszi, hogy a gyártórendszer modelljét egyszerűsítsem, ne kelljen az egész gyártórendszert bemodellezni, csak a robot számára a korábban már említett fontos részeket. Ezáltal a gyártórendszer és a mobil robot kapcsolata egy egyszerűen vizsgálható modellbe kerül, mellyel a robotok jóságát (pl.: kihasználtság, bekerülés, megtérülés) lehet minősíteni az egyes feladatokra nézve.

A kutatás fő motivációját a mobil robotok elterjedésének növekvő üteme adja, és az a tény, hogy a gyártórendszerekhez illesztett mobil robotok kiválasztása szimuláció segítségével történik, aminek több nehézsége is van, úgymint: rendkívül időigényes; magas szaktudás szükséges hozzá; drága szoftver, vagyis szimulációs környezet pl.: Plant simulator, Visual Components. A cél egy egyszerűbb modell megalkotása, amely a mobil robot perspektívájából írja fel a gyártórendszert így gyorsan számolható, és az eredmények gyorsan megkaphatóak. A következőkben egy elméleti koncepciót ismertetek, amit több gyakorlati esettanulmány egészít ki.

A felvázolt módszer alapján leírt esettanulmányok igazolják, hogy a mobilrobot központú gyártórendszer módszere a gyakorlati esetek egyszerű leírására alkalmas, és az egyszerűsége miatt másodpercek alatt számolható. Ezáltal ez a koncepció egy hatékony eszköz lehet a jövőben a gyártórendszerekhez illesztett mobil robotok modellezésére.

A módszer alapján a jövőben vizsgálható egyszerűbb esetekre a mobil robotok megtérülése, kihasználtsága, szükséges akkumulátor kapacitása. A módszer segítségével felírható modellek alapján az optimális robot kiválasztása jóval gyorsabb is lehet.

A gyártórendszerek felírása számos módon lehetséges, de ezek nem feltétlenül optimálisak a mobil robot szempontjából. Ezért arra kell törekedni a modell felírásakor, hogy a mobil robot három legfőbb feladata (feladatkiosztás, útvonaltervezés, forgalomirányítás) szempontjából egyszerűen kezelhető rendszert kapjunk. Többféle felírás közül is választhatunk, a két legjobban elterjedt megoldás: a diszkrét eseményű rendszer és a lineáris programozás. A bemutatott átfogó modellezési módszer elsősorban a diszkrét eseményű modellezési lehetőségekkel való megvalósítást támogatja. Ez a bonyolultabb modelleket is kezelni tudja, és optimális megoldást ad belátható idő alatt. A módszer egyes részei az alábbiak: a mobil robotok felírása, a gyártórendszer jellemzők modellezése, a mobil robot feladatok felírása.

Tekintsük az „ R ” halmazt a lehetségesen választható robotok halmazának, elemszáma legyen: „ l ”.

Minden egyes „ R_i ”-nek lehetséges „ j ” számú tulajdonsága: „ R_{ij} ”.

Ezáltal a robotok tulajdonságai megnevezhetők és kezelhetők. Néhány robot tulajdonság: robot maximális sebessége, teherbírása, méretei, kültéri vagy beltéri, elforduláshoz szükséges helye, akkumulátor kapacitás, raklap szállíthatóság, fel tud-e venni a földről rakományt, navigálás fejlettsége, a rakomány maximális fizikai méretei. A felsorolt tulajdonságokból nem mindegyiket vettem figyelembe az esettanulmányban.

A gyártórendszerben lévő gépek halmaza legyen „ M ”. A darabszámuk: „ k ”, egyes tulajdonságuk: „ l ”, az egyes gépek egyes tulajdonságát jelöljük „ M_{kl} ”-lel. Néhány lehetséges tulajdonság: ciklusidő, adott idő alatt kibocsátható darabszám, szerszámok száma, berendezések mérete, csatlakozási pontok száma, helye és helyzete, a munkafolyamat leírása.

A gyártórendszer jellemző paraméterinek is legyen egy halmaza, ezeket jelöljük „ S ”-el, darabszáma legyen: „ n ”. Az itt felsorolt paraméterek pl.: a gépek száma és helye, az állomások közötti útvonal hossza és szélessége stb.

A felírható modellek: diszkrét eseményű rendszerek, véges állapotú automaták, Petri-hálók.

A célfüggvények lehetnek:

- Kiválasztani a legolcsóbb robotot a feladatra.
- Kiválasztani a leggyorsabb robotot a feladatra.
- Minimalizálni az összes költséget.
- Maximalizálni a kihasználtságot.

A közbenső célfüggvények lehetnek arra vonatkozóak, hogy:

- Minimalizálják a torlódások miatti állásidőt.
- Elkerülik a holtponthoz.
- Optimalizálják az akkumulátor kapacitást.
- Teljesítmény növekedés a feladatok végrehajtásában.

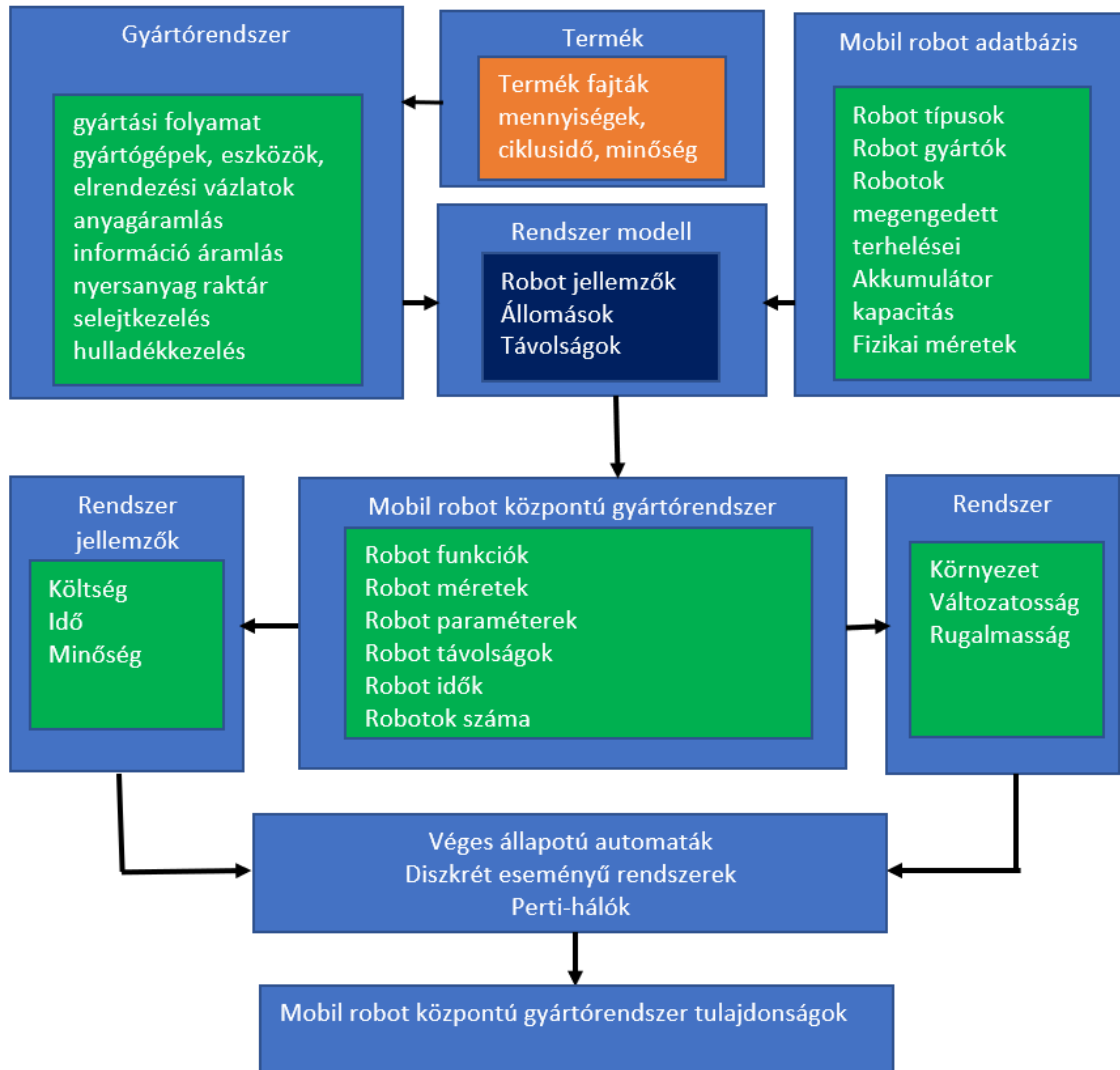
A 16. ábrán látható mobil robot központú gyártórendszer áttekintő ábrája. Az ábra leírása a következő:

Először a termék jellemzőit elemezzük, a termékek típusait, számát, a ciklusidőt és a minőségi jellemzőket. Ezek alapján készítjük el hozzá a gyártórendszer koncepciót. Ebben szerepel a folyamat, a gépek, az eszközök, az elrendezési vázlat, az anyag és információ áramlás. Benne van még az alapanyag és a hulladék kezelésén kívül a selejtek kezelése is.

A modell rendszer létrehozása szempontjából a gyártórendszer leírásán kívül a mobil robotok adatbázisa szükséges, ami tartalmazza a robotok típusait, gyártóját, terhelhetőségét, akkumulátor kapacitást és fizikai méreteit is. Ezekon kívül a modell rendszer felírásához felhasználjuk még a robot tulajdonságait, az állomások és azok távolságának adatait is.

Következhet a mobil robot központú modell felírása, amelyben a robot szempontjából releváns funkciók, méretek, paraméterek, távolságok, idők és darabszámok kapnak helyet. Itt kell definiálni minden fontos jellemzőjét a modellnek. A korábban felírt jellemzők alapján fel kell írni (nem számmal, hanem paraméterrel) az egyes állomások közötti távolságot, a robot sebességéből a távolságok megtételéhez szükséges időt. Figyelembe kell venni a robot által elvégzett feladatokat is, vagyis a robot funkciókat. A rendelkezésre álló tér

megjelenése a modellben bevisz egy újabb gyakorlati paramétert, amely a valós problémákra jellemző, vagyis a fizikai dimenziók figyelembevételét. Ha ez a modell összeállt akkor lehet a rendszer és a rendszer elemek paramétereit meghatározni. Az előbbire példa a környezettel vett kapcsolat, a változtathatóság és a flexibilitás, míg utóbbira a költség, az idő és a minőség.



16. ábra A mobil robot központú gyártórendszer modell áttekintő ábrája

Az így felírt modell célja, hogy az alábbi eszközökkel (legalább az egyikkel) kezelhető legyen, így készíthető szimuláció, amely kellően egyszerű ahhoz, hogy nagyszámú esetet is belátható idő alatt képes legyen lekezelni. Ezen eszközök a: Petri-háló, véges állapotú automata, diszkrét eseményű rendszerek.

Végül a felírt mobil robot központú modell megoldásából kinyerhetők a rendszer paramétereit és ezáltal a probléma megoldásai, amelyekből kiválasztható az adott szempontoknak megfelelő optimális megoldás.

A 16. ábrát figyelembe véve, és a lépéseken végig haladva elkészíthető a mobil robot centrikus modell felírás, és az említett három módszer egyikével elvégezhető a szimuláció. Majd a kapott eredmények értékelése alapján lehet továbblépni a folyamatban. Elfogadni az eredményeket, és ezek alapján folytatni a tervezést. Másik lehetőségként új paramétereket is figyelembe lehet venni, majd újra elkészíteni a modellt és a megoldásokat. Vagyis egy visszacsatolásos iteráció is lehetséges a végeredmény értékelése után.

A leírt gondolatmenetből megfogalmaztam a dolgozat első tézis értékű gondolatát.

1. Tézis:

Alkottam egy új modellezési módszert, amely a mobil robotot a gyártórendszerrel leválasztva képes kezelni. Ez a modell egy 10 állapottal és 8 átmenettel felrajzolt Petri-háló, amelyben a mobil robot szempontjából a gyártórendszer szükséges részei vannak megragadva, amelyek a következők: honnan, mit (fizikai méretek, súly, darabszám) és hova kell szállítani, milyen útvonalon, mekkora távolságra, hogyan tudja felvenni és lerakni a robot, és mekkora az akkumulátor töltés csökkenése a szállítási feladat hatására. A Petri-hálós modellt az akkumulátor kapacitás teszi korlátossá, minden állapot elérhető, és holtponmentes. Az új módszer neve a mobil robot központú gyártórendszer modell.

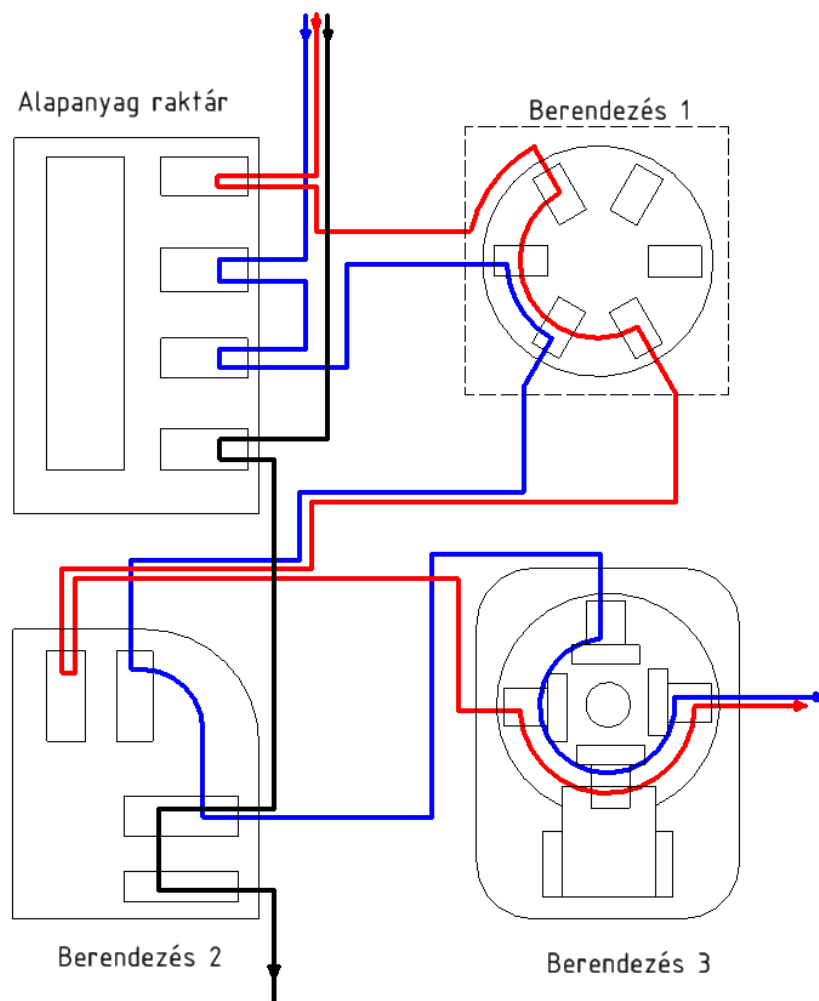
A tézist az alábbi elfogadott, publikálás alatt lévő publikáció támasztja alá:

M. Boleraczi, I. G. Gyurika: Data-Centric Engineering: Industrial mobile robot-based manufacturing system modelling potential

2.6 Esettanulmányok a gyártórendszer modellhez

A következőkben a korábban vázolt elméleti modell alapján esettanulmányokat dolgoztam ki, amelyek ezt a modellt használják.

Az ipari mobil robotok legfontosabb feladata az alapanyagok, félkész és késztermékek mozgatása a gyártórendszeren belül, a szigetszerű gyártóegységek (összekapcsolt, vagy egyedül álló gépek) között. A 17. ábrán az látható, hogy az egyes termékeknek különféle utakat kell bejárniuk a gyártórendszerben ahhoz, hogy késztermék legyen belőlük. Ezeket jelöltem különféle színekkel. A termékeket robotok szállítják az egyes állomások között.



17. ábra Gyártórendszer részlet vázlata háromféle termék útvonallal

Amikor a termék elkészül az egyik fázissal, akkor készen áll arra, hogy átszállítsák egy másik gyártógépbe. Erre az ipari mobil robotika jelenthet optimális megoldást. Az egyes szállítási feladatokat a robotokhoz lehet rendelni.

Az előző fejezetben bemutatott robot modell lehetővé teszi, hogy a robot paramétereit változtatni lehessen egy egyszerű szállítási feladat példáján keresztül. Ettől a modelltől azt várom, hogy a mobil robotokat egységesen lehet tárgyalni és összehasonlítani, ezáltal az adott feladatra optimalizálni a kiválasztásukat. Ehhez először egy olyan modellt alkottam meg, amely képes fogadni a robot paramétereit és feladatait, majd ezeket egy diszkrét eseményű rendszerben szimulálni és a kapott eredmények alapján következtetéseket tudok levonni a robot paramétereit és a gyártórendszer paramétereit között. Így optimalizálhatóvá válhat sok más mellett például az is, hogy egy adott darabszámhoz és úthosszhoz (két végpont távolsága) mekkora az optimális akkumulátor kapacitás.

Az első esettanulmány a feladatot még jobban egyszerűsíti és csak két pont és egy töltőállomás közötti mozgást vizsgál, majd a második esettanulmány kibővíti az állomások számát. Végül a harmadik esettanulmányban a modellt egy célfüggvény felíráshoz használom fel.

2.7 Két pont közötti szállítás

Az alapkonceptió szerint a feladatot úgy fogalmaztam meg, mint egy egyszerű szállítási feladatot, amely így épül fel: a robot két pont (legyen A és B) között valósít meg egyirányú szállítást. Vagyis „A” pontban felvesz egy munkadarabot (alapanyagot, vagy bármilyen más hasznos terhet, ezért a továbbiakban inkább „csomag” – ként hivatkozom rá, esetleg rakománynak fogom nevezni), majd ezt átszállítja a „B” pontba. Visszafelé az „A” pontba már nem szállít semmit, csak mozog. Az útvonalak (A-ból B-be és B-ből A-ba) hossza megadható, és lehet különböző érték is. Ezt egészítettem ki egy töltőállomással és a robot töltöttségével, ami már vizsgálható, és minden egyes pillanatban kiszámolható. Ezek lesznek a rendszer paramétereit. A python program a mellékletek között megtalálható a 118. oldalon.

A robot paramétereit az alábbiak:

- Akkumulátor kapacitás.
- Mozgás közben időegység alatti akkumulátor csökkenés (fogyasztás).
- Töltés ideje.
- Van-e csomag a roboton.

A gyártórendszer jellemzője:

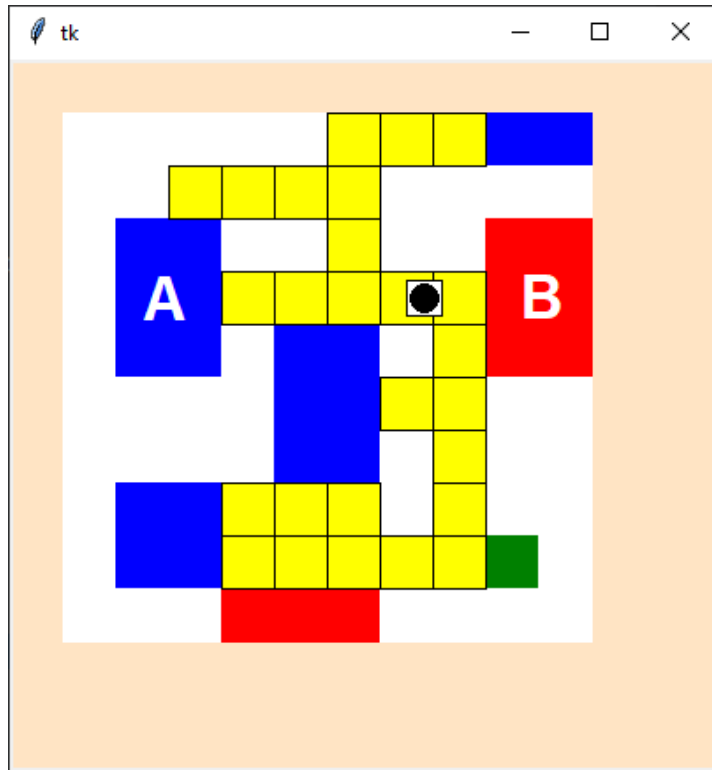
- Csomagok száma.
- A pont távolsága B ponttól.
- B pont távolsága A ponttól.
- Töltési pont távolsága B ponttól.

A felhasznált paraméterek a fenti paraméterekből kiszámolhatók, és ezek az alábbiak:

- c : csomagok száma,
- ab : mozgási idő $A \rightarrow B$ (szállítás van, csomaggal),
- ba : mozgási idő: $B \rightarrow A$ (mozgás van, csomag nélkül),
- t : töltési idő,
- t_m : töltőállomásra mozgás ideje

Így egy olyan modellt írtam fel, ami a fentieket kezelni tudja, ugyanakkor egyszerű és gyorsan lefuttatható. Az általános modellként felrajzolt Petri-háló alapján készítem egy állapotgép modellt, amely a robot különféle állapotait ragadja meg, és ezek változásait tudja kezelni. Ez az állapotgép modell látható a 18. ábrán.

elkészítettem pythonban a „ktinker” könyvtár segítségével, ez látható a 19. ábrán, a mobil robot a csomaggal fehér négyzetben fekete kör, az útvonalak sárgával míg töltési hely zölddel van jelölve.



19. ábra Kétpontos szállítási feladat szimulációja

A folyamat végén kapunk egy jelentést, ami leírja, hogy mennyi ideig tartott a szállítás és a töltés. Egy ilyen jelentésre mutat példát a 20. ábra, amelyben a durva kerekítések az időegységet hivatott szemléltetni, ti. perc \rightarrow óra és nap. Látható, hogy a csomagszám 500 db volt, és összesen 3544 percig végzett mozgást a robot, és 1913 percig állt. A töltések száma összesen 114 db volt. Az idő jelen esetben egységnyiként van definiálva, és minden esemény ehhez az időegységhez van igazítva.

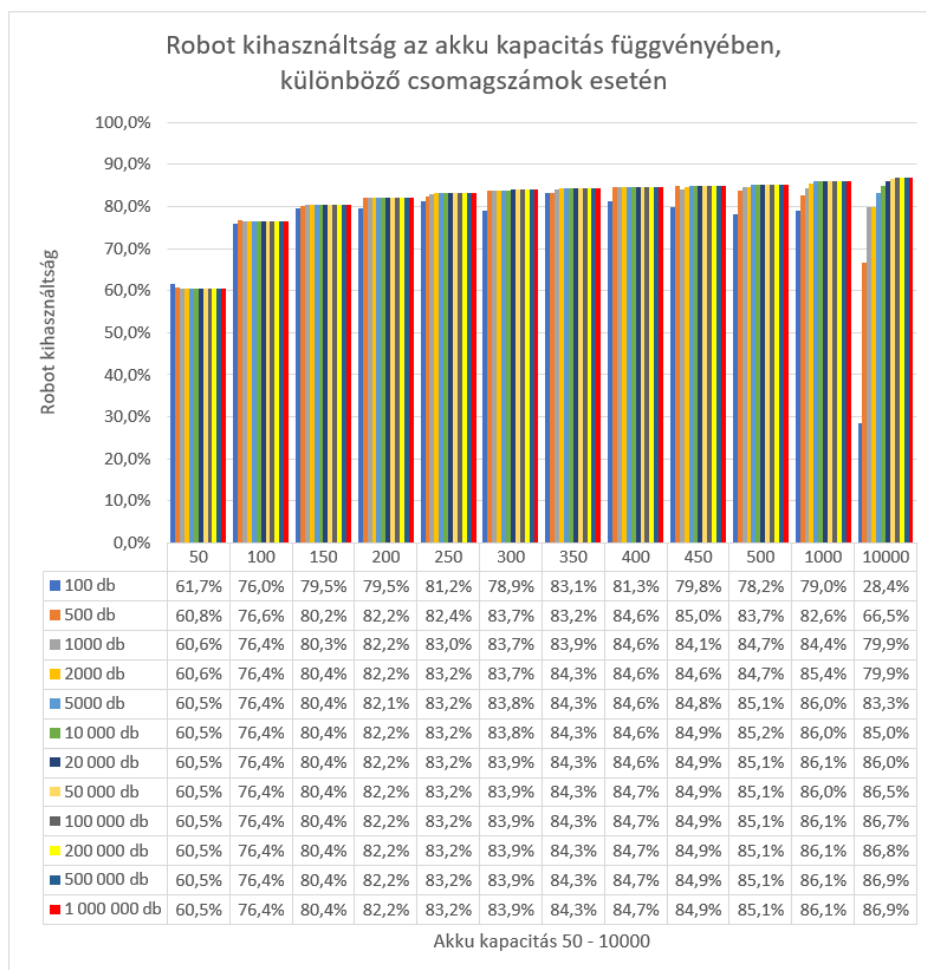
```
*****
**  Jelentés  **
*****
Átszállítva összesen: 500 db csomag.
A robot: 1913 percig állt, ami 31 óra, és 1 nap.
A robot: 3544 percig végzett szállítási feladatot, ami 59 óra, és 2 nap.
A robot kihasználtsága: 60.8% volt.
Összesen: 5826 perc telt el, ami 97 óra, és 4 nap.
Összesen 114 db töltés volt.
```

20. ábra A futtatott teszt eredménye szövegesen

Ezek alapján egy optimalizációs feladat lehet maximalizálni a kihasználtságot, amit az alábbi formában írtam fel a paraméterekkel:

$$\max K = \frac{\sum c * (ab + ba)}{\sum c * (ab + ba) + tm + t} \quad (40)$$

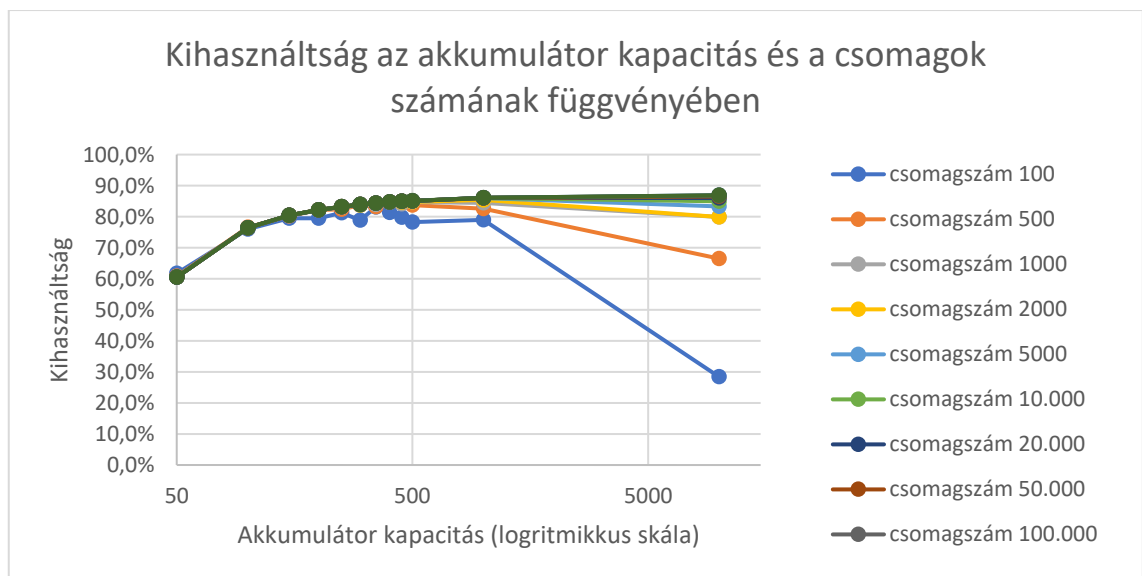
Az egyes paramétereket lehet változókét is kezelni, és az egész folyamatot ciklusba szervezni. Így megvalósítottam egy változó akkumulátor kapacitás és változó csomagszám paraméter szimulációt. Egészen nagy darabszámokat (1.000.000 db) is lehet szimulálni percek alatt. A robot szállítással töltött idejét arányosítva az eltelt idővel a robot kihasználtságát definiáltam, amit a ciklusba szervezett programmal ki tudok gyűjteni a különféle esetekre. Ennek az összetett elemzésnek az eredményét mutatja a 21. ábra.



21. ábra Az akkumulátor kapacitás és a csomagszámok összefüggése

Az ábrán a függőleges tengelyen a kihasználtság látható, a vízszintes mutatja az egyes akku kapacitásokat 50 – 10000 között. Az egyes oszlopok színei az átszállított csomagok számát mutatják 100 – 1.000.000-ig. Így leolvasható, hogy pl.: a 300-as kapacitáshoz és 100db csomaghoz tartozó érték (kék oszlop) alacsonyabb, mint a kisebb vagy nagyobb kapacitáshoz tartozó ugyanannyi csomagra vonatkozó kihasználtság. Tehát a 300-as kapacitás ennél a csomagmennyiségnél (és az 500db-nál is, narancssárga oszlop) egy rossz (nem optimális) választás lenne. Szintén leolvasható, hogy a 10000-es kapacitás esetén alacsony csomagszám mellett a magas töltési idő miatt túl alacsony a kihasználtság, viszont magasabb csomagszámok esetén is csak minimális növekedés érhető el az akku 10x-es méretének ellenére is. Az eredményeket egy olyan grafikonon is ábrázoltam, ahol az akkumulátor kapacitást logaritmikus skálán vettem fel, segítve ezzel az eredmények átláthatóságát, ezt mutatja a 22. ábra.

Az időegységet úgy értelmezzük, hogy 5 másodperc lesz 1 időegység. Ez pont elég arra, hogy valami egyszerűbb folyamat végbe menjen (pl.: dokkolás, munkadarab felvétel, 2D kód olvasás stb.), ugyanakkor jellemzően ennek többszöröseivel számolhatunk az egyes folyamatok idejére.



22. ábra Logaritmikus skálán ábrázolva az akku kapacitás

2.8 Többféle termék gyártására esettanulmány

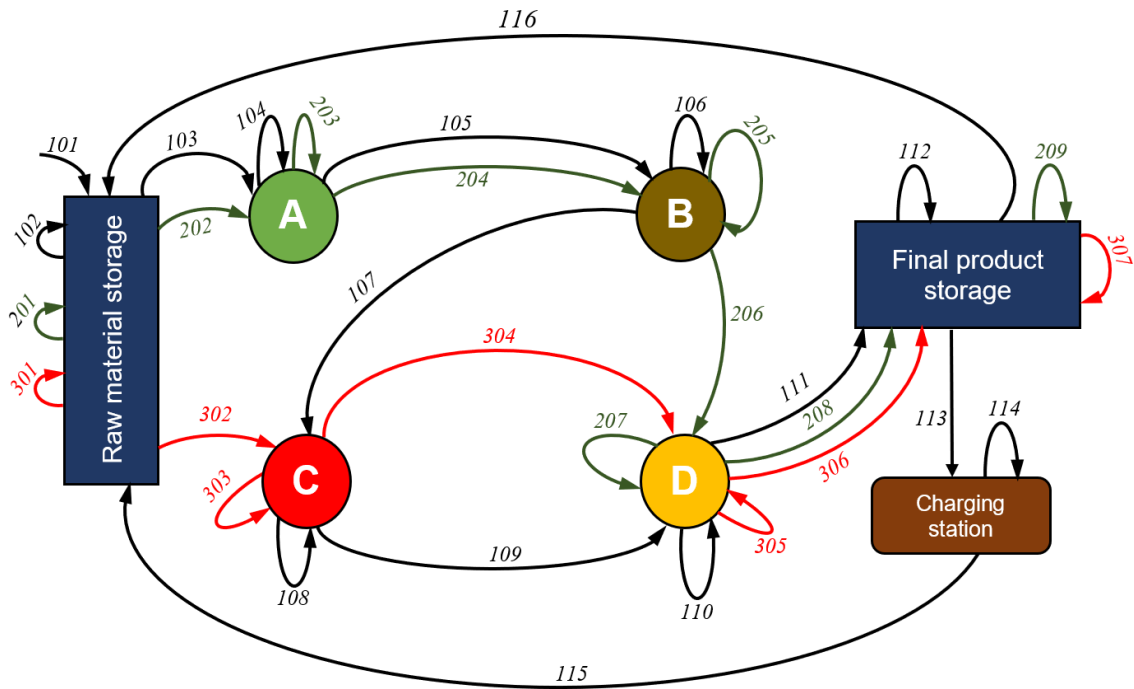
A második esettanulmányban kiterjesztésre kerül a modell, egy összetettebb feladatra. Növelni lehet az állomások számát és a termékek számát is. A

koncepció az alábbi: a robot az alapanyag raktárból kezdi a termékek gyártását. Innen kihozza a szükséges alapanyagokat, majd az egyes állomásokhoz megy, a terméknek megfelelően. Minden egyes állomáson eltölt egy bizonyos időt, majd megy a következőre. Az utolsó állomásról a készrű raktárba viszi a kész terméket. Itt ellenőrzi az akkumulátor kapacitást. Amennyiben az akkumulátorkapacitás egy küszöb érték alatt van, akkor szükséges a töltés, és a robot a töltő pontra megy, ahonnan töltés után visszatér az alapanyag raktárba. Ha nem szükséges a töltés akkor az alapanyag raktárba megy a robot.

Ebben az esetben 4 állomás van („A”, „B”, „C”, „D”), és számos különféle termék gyártására lenne lehetőség. Az egyes termékeknek át kell haladniuk állomásokon, de nem feltétlenül az összesen. A termékeket a gyártásuk során „meglátogatott” állomások alapján fogjuk értelmezni (elnevezni), így lesz „ABCD”, „ABD”, és „CD” termékünk. Minden állomáson van egy meghatározott idő ameddig a folyamat eltart, ez az egyszerűség kedvéért egységesen 5 egységben lett meghatározva, vagyis 25 másodperc. Mivel ezek is külön változóban vannak, ezért ezen értékek módosítása rendkívül egyszerű.

Így az egyes termékek gyártási ideje a szállítás nélkül: ABCD – 100 sec, ABD – 75 sec, CD – 50 sec. A teljes folyamat összes állapotát ábrázolja a 23. ábrán látható állapotgép, amelyben az ABCD termékhez tartozó állapotok 101-től 112-ig vannak. Mindhárom termék közös állapotai az akkumulátor töltéshez tartozó 113, 114 állapotok, valamint az alapanyag raktárba való visszatérést szolgáló 116 ill. 115 állapotok. Az ABD termékhez tartoznak a 201-209, a CD termékhez a 301-307 állapotok. Az állapotgépen ezúttal azt a módszert választottam, hogy az egyes állapotokat maguk a nyilak jelölik, míg a körök és téglalapok fizikai helyeket, vagyis magukat az állomásokat jelölik. Ezáltal az ábrázolás sokkal kompaktabb, ugyanakkor az értelmezhetőséget nem rontja.

A kidolgozott módszer gyorsaságára készített leghosszabb próba szimuláció több mint 5 millió darab gyártását foglalta magában, ami gyártási időben, ha átlagosan 75 sec-el számolunk, több mint 12 év, és ebben nincsen benne a robotos szállítások ideje. Ennek a programnak a futása kevesebb mint 9 perc volt.

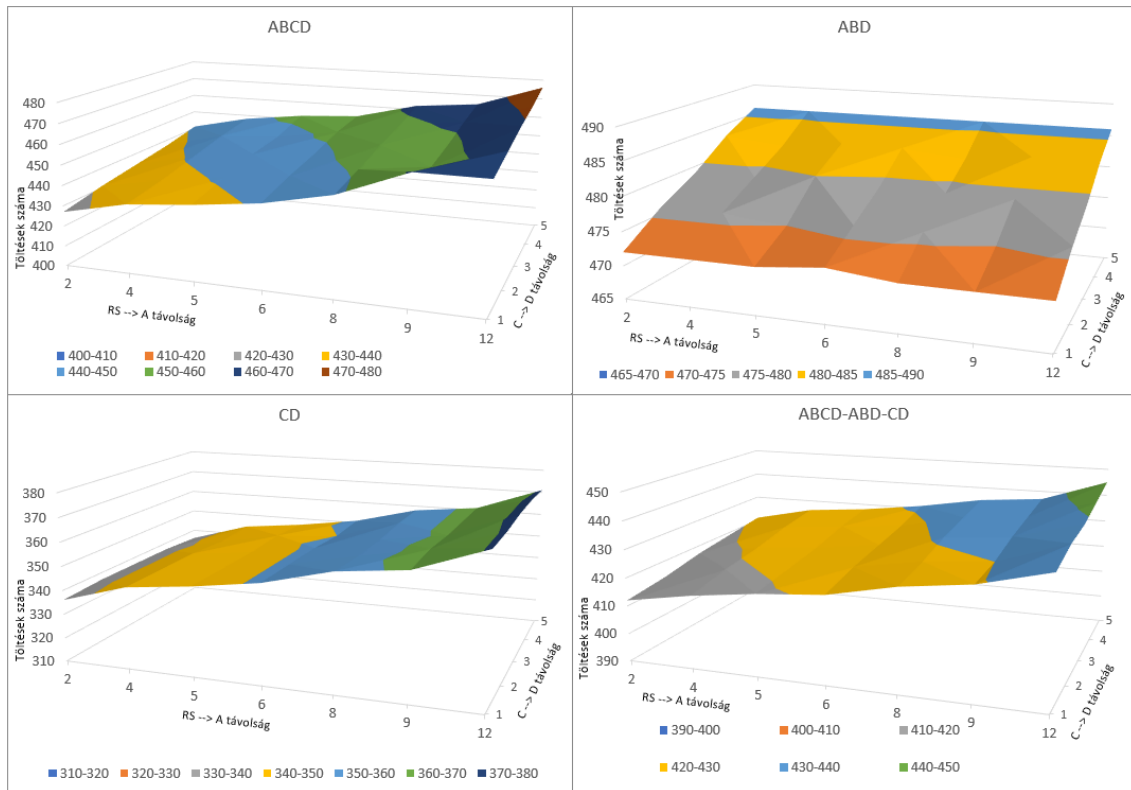


23. ábra Állapotgép többféle termék gyártására

A programmal megvalósítható az egyes gépek egymástól való optimális távolságának meghatározása is, az egyes gyártandó termékek száma alapján. Az optimalizációs kritériumot a robot töltésének számában jelöltem ki, mert ennek értéke az egész rendszer gyorsaságát is jellemzi, az előző esettanulmányban felfedezett összefüggések alapján. Így ezt a töltési számot akartam minimalizálni. Ugyanakkor megjegyzem, hogy a valódi gyártórendszer jellemzőkhöz közelebb állna a legyártott termékek darabszáma adott idő alatt. Ezt a harmadik esettanulmányban fejtem ki bővebben.

A gépek egymástól mért távolságát lehet optimális elrendezésben elhelyezni, amennyiben adott, hogy melyik darabból mennyit kell gyártani. Ezt több lépésben valósítottam meg, az eredményekből egy részt kiemelve a 24. ábrán látható grafikonokat készítettem el. A könnyebb ábrázolhatóság kedvéért csak két távolságot tekintünk. Ezek az RS-A (alapanyag raktár (Raw Material Storage - RS és az „A” pont) ill. a CD. Ezt a két távolságot ábrázoltam az alsó tengelyeken. A függőleges tengelyen a töltések száma látható. Három grafikon csak egy-egy termék gyártását mutatja, ezek rendre: ABCD, ABD, CD. Ezekből megállapítható, hogy a CD termék gyártása nem érzékeny az RS-A távolság változására, míg az ABD termék egyenletesen változik, és független a CD távolság változásától, viszont lineárisan függ az RS-A távolságtól. Ez csak egy egyszerű példa, a mobil robot központú gyártórendszer elmélete alapján leprogramozott kód ennél jóval

bonyolultabb összefüggések felderítésére is alkalmas, ezeket azonban nehezebb vizualizálni. Alkalmas lehet például nagyszámú útvonal lehetőség esetén is optimális megoldást keresni egy minimum kereséssel.



24. ábra Termékek gyártásának grafikonja a darabszám, a távolságok és a töltések számának összefüggésében

Előnyként elmondható, hogy az állapotgép alapján a program könnyen felépíthető, a felépített program jól átlátható, és bővíthető.

Hátrányként elmondható, hogy a programozói tudás mellett az állapotgépek elméletével is tisztában kell lennie a fejlesztőnek.

2.9 Kiértékelő függvényként való felhasználás

Harmadik esettanulmányként írtam egy programot, amivel az ABCD távolságokat lehet genetikus algoritmussal kiszámoltatni, egy 5 – 45 közötti 5 – ös osztásközű sorozatból, így generálva gyártórendszer variánsokat. A 23. ábrán látható állapotgép alapján írt python programot annyiban módosítottam, hogy egyetlen töltéssel legyártható darabszámot tudjak vele mérni. Így a három termékre, három darabszámot kaptam, amiket egy-egy tele töltéssel le tud gyártani a rendszer. Így a rendszer teljesítmény mérője a legyártott darabok száma lett. Ezeket a számokat összeadtam, így kaptam egy értéket, amit a gyártórendszer jellemző paraméterének tekintek, így ez adja meg az egyes

gyártórendszer variánsok teljesítményét. Végül a genetikus algoritmus kiértékelő fitness függvényeként használtam fel ezt a kódot, így szimulálva minden egyes egyedre a legyártható darabszámokat. Időben ez így hosszabb lehet, mint egyetlen képletet felhasználva kiértékelni az egyes egyedeket, viszont egy valós gyártórendszer működést ragad meg.

Kétféle programot készítettem, ami csak a kiértékelő függvényben tér el, az összes többi paramétere, a generációk létrehozása, a keresztezés, mutálás és kiválasztás teljesen azonos. A saját kiértékelő függvényemet a későbbiekben lehet akár tovább fejleszteni azzal, hogy az egyes termék változatokból (ABCD, ABD, CD) különféle mennyiségeket gyártunk, vagyis súlyozhatjuk őket.

Az első tesztnél program ezeket a paramétereket használtam:

- A populáció mérete: 10 egyed.
- Generációk száma: 100.
- Elitizmus: 1 egyed kiválasztva a szülők közül.
- Versenyszerű kiválasztás: 10 egyed.
- Keresztezés esélye: 80%.
- Mutáció esélye: 10%.

Lefuttattam a programot és a 16. táblázatban leírt eredményeket kaptam.

16. táblázat Genetikus algoritmus eredmények I.

Minimális feszítőfa algoritmussal			
	Legjobb fitnessz érték	konvergál	futási idő [sec]
Fitnessz összsúly	288	6. generáció	0,078
	358	22. generáció	0,07236
	309	11. generáció	0,06022
Mobil robot központú elv alapján			
	Legjobb fitnessz érték	konvergál	futási idő [sec]
Fitnessz összsúly	650	72. generáció	28,856
	650	34. generáció	33,117
	650	75. generáció	32,6376

Jól látható, hogy a minimális feszítő fa által adott fitnessz érték messze elmarad az én mobil robot központú gyártórendszer szemlélet alapján felírt kiértékelő algoritmusomtól.

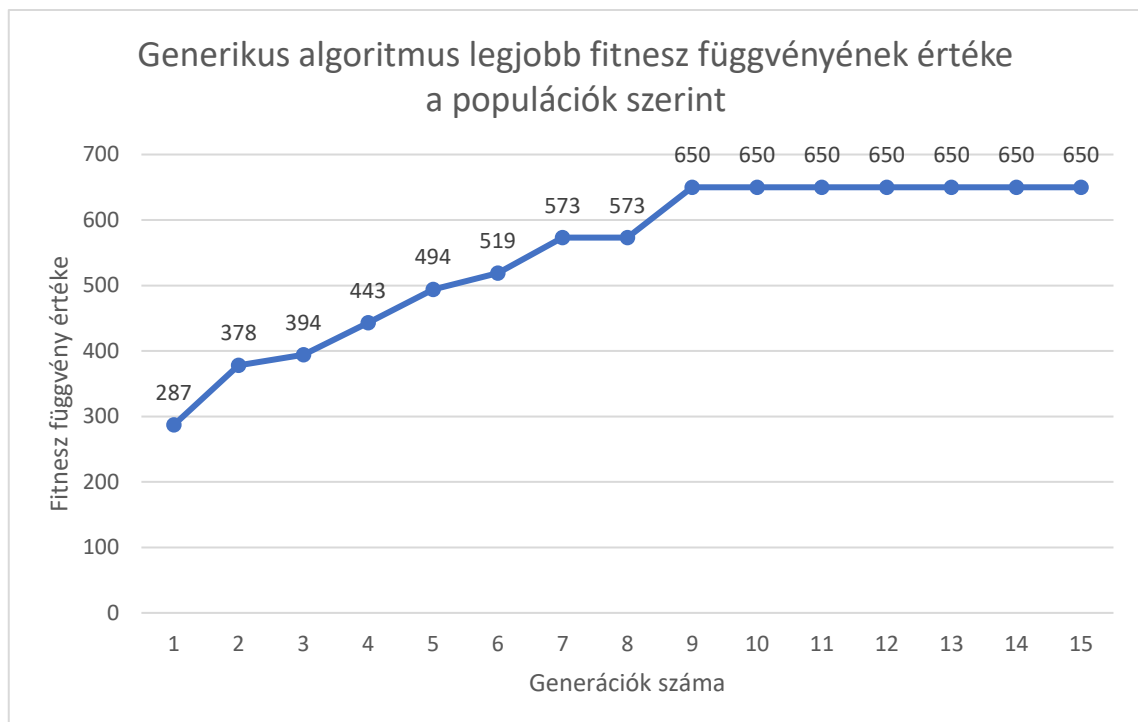
Nagyobb egyedszámot és kisebb populáció nagyságot választva:

- A populáció mérete: 100 egyed.
- Generációk száma: 50.
- Elitizmus: 2 egyed kiválasztva a szülők közül.
- Versenyszerű kiválasztás: 6 egyed.
- Keresztezés esélye: 80%.
- Mutáció esélye: 10%.

17. táblázat Genetikus algoritmus eredmények II.

Minimális feszítőfa algoritmussal			
	Legjobb fitnessz érték	konvergál	futási idő [sec]
Fitnessz összecsúly	286	3. generáció	0,1787
	363	4. generáció	0,2137
	458	4. generáció	0,169
Mobil robot központú elv alapján			
	Legjobb fitnessz érték	konvergál	futási idő [sec]
Fitnessz összecsúly	650	6. generáció	171,89
	650	9. generáció	172,24
	650	10. generáció	148,816

A fenti táblázat második részéből egyet kiragadva a 25. ábrán ábrázoltam a legjobb fitnessz értékeket a generációk számának függvényében.



25. ábra Fitnessz függvény értékek

Ezek alapján megállapítható, hogy a mobil robot központú gyártórendszer modellel felírt kiértékelő függvény 10 generáció alatt eljuttatta a genetikus algoritmus egyedeit az optimális megoldáshoz, míg a minimális feszítő fa algoritmus 3-4 generáció alatt beragadt, és nem találta meg a gyártórendszer teljesítménye szempontjából optimális értéket.

2.10 A módszer előnyei az esettanulmányok alapján

Ami megkülönbözteti ezt az egész metódust más kereskedelmi rendszerektől az az alábbi három dolog: Először, ez a gondolatmenet algoritmikusan közelíti meg a problémát, és nem igényel semmilyen iparban kapható drága szoftvert, csak alapvető programozói tudást. Másodsor, mivel egy csomag átszállítása 0,25 ezredmásodpercnyi futási idő, ezért extrém nagy darabszámok is gyorsan szimulálhatók, vagy extrém nagy esetszámot is lehet belátható időn belül szimulálni. Ehhez csak néhány paraméter átírása szükséges, amit akár egy összetettebb programban automatikusan is el lehet végezni, ezáltal automatizálva az optimumkeresést. Harmadsor, a szimulációba később be lehet vinni a valós körülményeket jobban közelítő elemeket is. Ezeket valószínűségi változókkal lehet kezelni, ami által lehetővé válik pl.: egyes robot típusok meghibásodásának valószínűségét a modell részévé tenni. Akár az elakadásokat, operátori segítséget is be lehet így vinni. A kiértékelő függvényként való felhasználás a lehetőségek kiterjesztését is mutatja.

A futási idők egyes gyártott darabszám esetében, rendre: 100.020 darab/23,9 másodperc; 1.000.020 darab /228,3 másodperc; 20.020 darab /4,59 másodperc; 504.020 darab /47,16 másodperc; 5.040.020 darab /533,54 másodperc.

2.11 Újdonság tartalom és hozzájárulás

A második tézis két részből tevődik össze:

A 2. tézis „A” része: A mobil robot központú gyártórendszer szemléletét felhasználva igazoltam, hogy egy egyirányú, két pont közötti szállítási feladatra a robot akkumulátor kapacitása két nagyságrendet is átölelő elméleti határok között vizsgálható, és az ideális méret a feladat függvényében erőforráshatékonyan meghatározható, tehát a módszer jól skálázható.

A 2. tézis „B” része: Igazoltam, hogy a mobil robot központú gyártórendszer modellel felírható olyan számítási modell, amely magasabb szintű optimalizációs algoritmusok kiértékelő függvényeként használható. Az elvégzett esettanulmányban a genetikus algoritmus egyedeit kiértékelő fitness függvényként való felhasználás a valós gyártási folyamatokat veszi figyelembe és így ad jellemző paramétert, ami emiatt a kísérletek alapján 10 generáció alatt elérte az optimális értéket.

A tézist az alábbi elfogadott, publikálás alatt lévő Q1-es publikáció támasztja alá:

M. Boleraczi, I. G. Gyurika: Data-Centric Engineering: Industrial mobile robot-based manufacturing system modelling potential

3 A szakértői rendszer felépítése

Az dolgozat első részében ismertetett szakirodalmat röviden áttekintve elmondhatjuk, hogy a mobil robotok integrációja számos területet felölel, és számos megoldást is kidolgoztak rá, ugyanakkor az egyes gyártórendszerekhez illő optimális robot kiválasztásának támogatására szakértői rendszert még nem publikáltak. Jelen kutatás egyik célja egy ilyen szakértő rendszer tudásbázisának létrehozása és egy tesztkörnyezetben való tesztelése.

3.1 Az optimálási feladat megfogalmazása

A korábban áttekintett gyártórendszerek kiegészítése vagy része lehet az ipari mobil robot. Ezeket a robotokat a versenyképesség megőrzése, ha lehet növelése mellett kell a gyártórendszerekbe integrálni. Ezt úgy lehet elérni, ha a gyártórendszer működése a robottal kedvezőbb lesz, mint a robot nélkül, vagy adott esetben az egyik robotot összevetve a másikkal a gyártórendszer minőségi jellemzőiben javulás tapasztalható.

Ezek alapján az optimálási feladatot úgy fogalmazom meg, hogy: **Olyan robot kiválasztása a cél, amellyel a gyártórendszer jellemzők teljesítmény értékei a maximálisak lesznek, ugyanakkor a gazdasági szempontokat szem előtt tartva a karbantartási, a beruházási és üzemeltetési költségek minimálisak lesznek.** A feladat peremfeltételei:

- A gyártórendszer elérhető paraméterei, teljesítmény jellemzőihez szükséges értékek, paraméteresen, vagy számértékkel, becsült vagy mért értékei.
- A gyártórendszer felépítése, gépek és berendezések fajtái, száma, ezek távolsága, elrendezési vázlat.
- A kiválasztható robotok halmaza. pl.: a minimális és maximális teherbírás alapján a legkisebb terhelhetőségűtől a legnagyobb teherbírású robotig.
- A választható robotokról elérhető összes publikus információ, mint a robotok tulajdonságai.
- Az emberi szakértők által képviselt tudás, pl.: robot üzemeltetési tapasztalatok alapján.

A célom tehát egy olyan eszköz fejlesztése, amely ezt az optimális kiválasztást támogatja, segíti, ehhez érdemben hozzájárul [83].

3.2 Tervezési paraméterek a mobil robotok integrálásához

Ahhoz, hogy választani tudjunk a rendelkezésre álló seregnyi mobil robotból ismernünk kell azok egyedi jellemzőit, másrészt a mobil robotokat egymással összehasonlíthatóvá is kell tenni. A 26. ábrán néhány mobil robot, típusát és képét láthatjuk.

OMRON LD-60	OMRON LD-90	OMRON LD-250	SEER AMB-150	SEER AMB-300	SEER AMB-300XS
					
MiR100	MiR250	MiR600	SEER SFL-CDD16	AGILOX ODM	Toyota SAE-160
					

26. ábra Néhány piacon elérhető mobil robot

A szakirodalmakból az alábbi tervezési paramétereket tekintetem át:

- Robotok száma.
- Robotok sebessége.
- Robotok gyorsulása az adott sebességre.
- Robotok lassulása az adott sebességről.
- Csomag felvételi idő.
- Csomag lerakási idő.

A korábban elemzett cikkek alapján ezek a kritikus faktorok:

- AGV torlódás (%) (Congestion), amikor haladhatna a robot, de nem tud, mert az útvonala blokkolva van.
- AGV kihasználtság (%) (Utilization): 1 – a teljes szimulációs idő százalékos aránya, amelyet a járművek a parkolóhelyre való eljutással és a parkolással töltenek.
- Átbocsátási teljesítmény (Throughput): időegység alatt mennyi a legyártott (átengedett) darabszám.

Mások a gyártórendszer jellemzőket ragadják meg az alábbiakban:

- Tétel méretek.
- Az érkező munkák gyakorisága.
- A munkák időtartama.
- Gépek száma.
- Folyamatok száma.
- Átbocsátási képesség.
- Munkadarab csere ideje.
- Pufferek mérete.
- Termék variánsok száma.
- Távolságok a gépek között.
- Munkadarabok útvonala.
- Termékek száma.

A mobil robotokat jellemző paramétereiket a gyártórendszer jellemző paramétereivel is össze kell hangolni, pl.: adott rakomány méretet és súlyt a robot elbírná és a megfelelő gyorsasággal tudja mozgatni. Ezen összehangolási folyamat egyik megoldása, ha szimulációkat futtatunk az egyes gyártórendszerekre. Ezeknek a szimulációknak az összeállítása elég magas szakértelmet igényel. Viszont a fenti rendszerparaméterek közül számos paramétert ki lehet belőlük olvasni, és így a gyártórendszer optimális működését (az előre felvázolt esetekre un. „szcenáriókra”) garantálni lehet.

3.3 Különbségek a robotok között – adatbázis

A mobil robotokat egy adatbázisban szükséges letárolni, így összehasonlítható paraméterek szerint lehet ezeket kezelni. A 18. táblázatban látható egy ipari mobil robot adatbázis részlete.

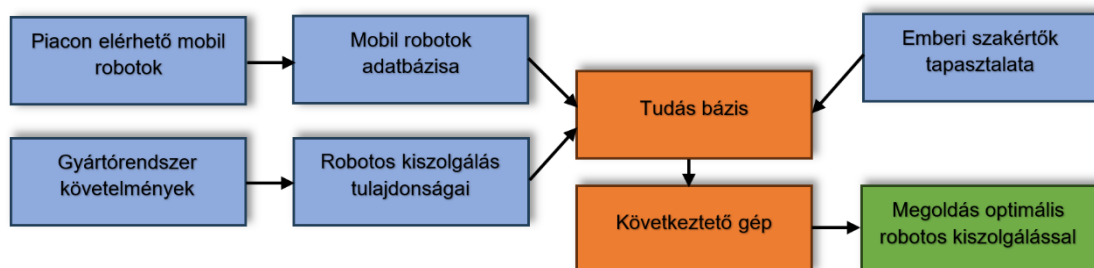
18. táblázat A mobil robotok adatbázisa

Industrial mobile robots database														
Robot No.	Robot manufacturer	Robot type	Robot loading capacity, kg	Inside	Outside	Weight of the robot	Battery capacity	Time of battery charging	Towing_ option	Loading from the ground available	Maximal velocity	Pallet_ moving	Useable in ESD area	Cleanroom option
1	MiR	100	100	True	False	77	8	1	False	False	5	False	False	False
2	MiR	250	250	True	False	97	6	0,8	False	False	5	False	False	False
3	MiR	600	600	False	False	243	7	0,6	False	False	5	False	False	False
4	MiR	1350	1350	True	False	247	8	0,8	False	False	5	False	False	False
5	Robotnik	RB-Robout	1000	True	False	350	6	0,5	False	False	7	False	False	False
6	Robotnik	RB-Theron	200	True	False	45	6	0,5	False	False	7	False	False	False
7	OMRON	LD-60	60	False	False	62	5	0,4	False	False	4	False	False	False
8	OMRON	LD-90	90	True	False	62	5	0,5	False	False	4	False	False	False
9	OMRON	LD-250	250	True	False	148	4	0,4	False	False	4	False	False	False
10	OMRON	LD-1500	1500	True	False							False	False	False
0														

A robotok közötti különbségek egy része könnyen számszerűsíthető. Ide tartozik a teherbírás vagy a méret. Ezeket a mobil robot gyártók általában megadják a robotok adatlapján. A 18. táblázat a hivatalos adatlapon elérhető és könnyen számszerűsíthető adatokat tartalmazza. Ezeket az adatokat felhasználva hozzuk létre a tudásbázist, amely folyamat a tudásreprezentáció szempontjából a leginkább lényeges. Az emberi tapasztalat (tudás) megragadására, a folyamatmérnöki, gyártósori mérnöki szakértői tudás reprezentálására a fenti táblázat kiegészítése lehet.

3.4 Különbségek a robotok között – tudásbázis

A robotok közötti különbségekből kezdjük el felépíteni a tudásbázist. A szakértői rendszer felépítése szerint áll egy tudásbázisból a szabályokból és a következtető gépből. A kutatás ezen része a tudásbázis felépítésére vonatkozik. Az adatok között kapcsolatokat akarunk definiálni, amelyekből létrehozunk a tudásbázist. Ezen a tudásbázison tudunk műveleteket végezni pl.: a következtető gép segítségével, de más módszerek is szóba jöhetnek. A felvázolt rendszer működése a 27. ábrán látható.



27. ábra A szakértői rendszer munkafolyamati ábrája

Először összegyűjtjük a piacon elérhető mobil robotokat, és adatbázisba szervezzük őket. Ezzel párhuzamosan a gyártórendszer követelményeit is feldolgozzuk és a robotra vonatkozó követelmény listát állítunk össze. A humán

szakértők tapasztalata nehezebben megfogható része a rendszernek. Itt olyanokra gondolunk, mint:

- Adott típusú robot a paraméterek alapján alkalmas lenne a feladatra, viszont hasonló feladatot csak nagyon sok hibával, vagy egyáltalán nem tudott megoldani. Ez értelmezhető egy „megbízhatóság” értékelésnek is.
- Adott robot nem lenne alkalmas a feladatra (pl.: teherbírás korlát) viszont tapasztalat szerint a gyakorlatban tökéletesen el tudja látni az adott feladatot.
- Adott gyártórendszerben hogyan teljesítettek a robotok.

Ez a három tényező együttesen adják meg a tudásbázist. A következtető gép, vagy más módszer segítségével pedig a feladatnak leginkább megfelelő robot kiválasztható.

A tudásbázis többféleképpen létrehozható: ítéletlogikával, elsőrendű logikával, temporális logikával, FUZZY módszerrel és valószínűségi logikával. Az elsőrendű logikában a világ objektumokból és ezek közötti relációkból áll. Az ítéletlogikában minden egyes tényhez kell egy logikai változót rendelni, aminek értéke lehet igaz vagy hamis. Az első lépésben a cél ezeket a logikai változókat automatikusan előállítani az adatbázisból. Az egyes tulajdonságokhoz tartozóan hoztam létre a változók fajtáit. Itt meghatározott értékeket tekintettem adott osztályokhoz tartozónak.

Az adatbázisban lévő adatoknak két alaptípusa van:

- Logikai értékkel jellemezhető adatok.
- Számértékkel jellemzett adatok.

A logikai értékkel jellemzett adatokat könnyen lehet az ítéletlogikában értelmezni. Egy ilyen változót úgy képeztem, hogy a robot gyártóját és típusát, esetleg típusszámát összevontam, és ehhez még hozzávettem az adott tulajdonságot. Így összeáll a változó neve, amivel már tudunk az ítéletlogikában tovább dolgozni. Ez a változó így már kaphat logikai értéket, ami az adatbázisban található értéknek feleltethető meg.

Így megkaptam a logikai változók egy részét, amelyek az adatbázisból kinyerhetők: robot sorszáma, robot gyártója, robot típusa, robot súlya, robot teherbírása, beltéri, kültéri, robot akkumulátor kapacitása stb.

Ezután összeállítottam egy változó nevet, azután az értékét beállítottam az adott mezőben található értékre. Az egyik gyakorlati kérdés, hogy a változó nevét hogyan tudjuk összeállítani úgy, hogy azzal a programban, mint változó névvel dolgozni lehessen.

Az ítéletlogika szimbólumokkal dolgozik. Ezekből a szimbólumok lesznek a változók. Ezeket szövegesen kell definiálni. Az alábbi kérdésekre kell választ adni a módszernek ebben a lépésében:

- Milyen tulajdonságokat keresünk?
- Milyen tulajdonságok vannak a táblázatban?
- Milyen tulajdonságokat emelünk ki ezek közül?
- Ezekből milyen változókat hozunk létre?
- Milyen logikai változók lesznek a tudásbázisban?

19. táblázat A tudásbázis részlete

	robot0	robot1	robot2	robot3	robot4	robot_searched
robot_size_small	True	False	False	False	False	False
robot_size_medium	False	True	True	False	True	True
robot_size_large	False	False	False	True	False	False
robot_capacity_weak	False	False	False	False	False	False
robot_capacity_medium	True	True	False	False	True	True
robot_capacity_heavy	False	False	True	True	False	False
robot_battery_capacity_low	True	False	True	False	False	False
robot_battery_capacity_medium	False	True	False	True	False	False
robot_battery_capacity_high	False	False	False	False	True	True
robot_outside	False	False	False	True	False	False
robot_inside	True	True	True	True	True	True
robot_pallet_moving	False	False	False	False	False	False
robot_towing	False	False	False	True	False	False

A 19. táblázatban látható a fentiek alapján összeállított tudásbázis.

Az alábbiakban egy példa a változók nevének összeállítására: robot + sorszám + tulajdonság fajta + tulajdonság részlet:

Robot_sorszám_tulajdonságfajta_tulajdonságrészlet – robot1_size_large.

Így a logikai értékeket tartalmazó mezők az adatbázisból könnyen létrehozhatók.

A számértékekkel rendelkező mezők létrehozása már nehezebb. A példa kedvéért tekintsük a „Battery capacity” cella értéket a tudásbázisban. Ez egy számértéket tartalmazó paraméter, amit szintén le kell kezelni az ítéletlogikában. Két megoldást gondoltam végig erre. Az első magától értetődő megoldás az, hogy minden egyes számértékhez felveszek egy logikai változót, vagyis minden egyes értékhez egy saját új változót hozok létre. Erre egy példa: akkumulátor kapacitás esetén 8 db értékhez, 8 logikai változót veszek fel minden egyes robothoz. A sok változó miatt ez egy erőforrás pazarló megoldásnak tekinthető.

Másik lehetőségként mivel a számértékekhez logikai értékeket kell rendelni, megfelelően csoportosítani lehet az egyes értékeket, jelen esetben három csoportba. Így a Battery capacity értéke lehet: low, medium, high. Az egyes logikai értékeknek megfelelő számértékeket természetesen a program számára deklarálni kell, pl: low = 0,5 – 3 között, medium = 3,1 – 5,5 között, high = 5,6 – 8 között. Ezt egy FUZZY jellegű érték is jellemezheti, ahol az egyes határok nem ennyire élesek, hanem „átfednek” egymásba.

Látható, hogy ezzel a módszerrel az adatbázisból előállítható a tudásbázis, majd azon végrehajthatók a további műveletek, így kapva meg az optimális robotot. A módszer használatát két esettanulmányon is prezentálom.

3.5 Szakértői rendszer első esettanulmánya

Az alábbiakban ismertetek két esettanulmányt, amelyben az ismertetett módszert alkalmazom lépésről lépésre.

A tudásbázis létrehozásához és a kiértékeléshez használt program „logic.py” könyvtára megtalálható a „CS50’s Introduction to Artificial Intelligence with Python” ingyenes online kurzuson az edx.org címen.

Kiindulásként a 18. táblázatból látható adatbázisból vettem ki egy kisebb részletet, ami a 20. táblázatban látható.

20. táblázat Az adatbázis egy részlete

Name	Description	Variable type	Example value
Inside	inside or not	boolean	True
Loading capacity	weight [kg]	integer	20
Battery capacity	time [minute]	integer	120
Pallett moving option	available or not	boolean	True
Towing option	possible or not	boolean	False

Elsőre nézzük meg a logikai változók beolvasását és használatát. A logikai változókat átalakítjuk az ítéletlogikát használó program számára szimbólummá.

- `robot_10_Inside = Symbol('Robot 10 Inside')`

Majd ezt hozzáadjuk a tudásbázishoz, attól függően, hogy az értéke True vagy False.

Amennyiben az értéke „True” akkor így:

- `knowledge.add(robot_10_Inside)`

Amennyiben az értéke „False” akkor így:

- `knowledge.add(Not(robot_10_Inside))`

Ahhoz, hogy az egyezést is jellemezni lehessen újabb szimbólumokat kell felvenni. Ezek azt jelentik, hogy az adott számú robot adott tulajdonsága egyezést mutat a követelményekkel. Ezeket az alábbi formában írjuk fel:
`robot_1_Inside_fit = Symbol("Robot 1 inside fit with the requirement")`

Azért, hogy ezeket automatikusan lehessen generálni készítettem egy program-részletet, ami a táblázatból kiolvasott értékeket a fenti mondat alakjába rendezi. A `row_number` változóban a táblázat hossza által megadott szám szerepel, vagyis az adatbázisban található robotok száma.

Ciklus 3-tól a sorok számáig, i változóval:

a robot_number legyen egyenlő a táblázat i-edik sor, 1. oszlopának értéke

a parameter_id legyen egyenlő a táblázat második sor, 17. oszlopának értéke

írasd ki (("robot_" + str(robot_number) + "_" + str(parameter_id) + " = Symbol("Robot " + str(robot_number) + " " + parameter_id + ""))

A fenti ciklussal az összes robot egy adott paraméter jellemzője beolvasható, és a szimbólum létrehozásához szükséges mondat előállítható. Az utolsó sorban állítom elő a szimbólumot. Következő lépésben a szimbólum értékét kell beolvasni, és hozzárendelni a szimbólumhoz. A program ezeken a szimbólumokon végez műveleteket. A tesztelést az első implementált verzióban egy teljes igazságtáblázat felépítése alapján végeztem. Emiatt az első esettanulmányban összesen 29 szabályt alkalmaztam, így a kiértékelés ideje 937.72 sec. Minden egyes új szabály a duplájára növeli a kiértékelés idejét, amit a programmal elvégzett mérési eredmények vissza is igazolnak, így 24 szabály esetén 31,55 sec míg 23 esetén 15,66.

A tesztek alapján a módszer működőképes, a tudásbázisban szereplő robotok közül ki tudja választani azt a robotot amelyik megfelel a követelményeknek. Ez a 19. táblázatban látható robot4, mert ott van a tulajdonságokban a legtöbb egyezés a keresett robottal.

Az esettanulmány igazolja, hogy a vázolt módszer alkalmas arra, hogy egy adatbázisból egy tudásbázist állítson elő, amit ki lehet értékelni. A tesztek a kiértékelési módszer egyszerűsége miatt viszonylag kevés szabállyal történtek. A szabályok számának növelésre, és ezáltal valós méretű problémákra való alkalmazhatóságára a módszernek, további kutatási lépések szükségesek. Ehhez másfajta kiértékelési metódus szükséges, hogy a futási idő ésszerű keretek között maradjon. Másik lehetőség, hogy a kiértékelést két lépcsőben kell végrehajtani, az első lépcsőben a szabályok számát kell csökkenteni, a feladat valamely kritikus paraméterének nem megfelelő robotok gyors kizárásával.

3.6 Szakértői rendszer második esettanulmánya

A CLIPS környezetben két esettanulmányt készítettem el. Először egy egyszerűbbet, a folyamat megértése, és gyakorlása miatt. Ebből látható az alábbiakban két tény és egy összetettebb szabály leírása.

```
(assert (robot1-size-small))
```

```
(assert (req-robot-size-small))
```

```
(defrule robot_1-size ""
```

```
  (or (and (robot1-size-small)(req-robot-size-small))
```

```
  (and (robot1-size-large)(req-robot-size-large))
```

```
(and (robot1-size-medium)(req-robot-size-medium)))  
=>  
(assert (robot1-size-good))  
(printout t "Robot 1 size fit with the requirement.")
```

A fenti programkód szintén megíratható python kóddal is. A CLIPS működése lehetővé teszi a szabályok által vezérelt működés révén a rendkívül gyors futást. A szabályok között versengés van, hogy melyik szabály tüzelhessen (vagyis fusson le a következőnek). Erre többféle stratégia is van, én azt használtam, hogy a szabályoknak van egy prioritása (saliency), ezt egy értékkel állítjuk be minden szabálynál. Az így elkészült teszt azt mutatta, hogy a futás jelentősen felgyorsult, ahogyan az várható is volt.

3.7 Szakértői rendszer harmadik esettanulmánya

A fejlesztés utolsó szakaszában egy kibővített megközelítést használtam, ebből állt össze a harmadik esettanulmány. A robotok tulajdonsága mellett a szakértői tudást is felhasználtam, ill. a robotok gyártórendszeri használata közben felgyülemlett tapasztalatokat, gyártórendszer jellemzőket, valamint megbízhatósági és teljesítmény mutatót is. A program kódot angolul írtam a publikációs lehetőségeket figyelembe véve, de a dolgozatban magyarul is szerepelnek a tulajdonságok, melyeket az alábbiakban adok meg, a robotok katalógusban elérhető adataira nézve:

- azonosító, robot_id,
- gyártó, manufacturer,
- típus, type,
- terhelhetőség, loading_capacity,
- beltéri, inside,
- kültéri, outside,
- robot tömege, robot_weight,
- akkumulátor kapacitás, battery_capacity,
- töltés ideje, time_battery_load,
- kiskocsi húzási opció, towing,
- felvétel a földről, ground_load,
- maximális sebesség, max_velocity,
- raklap mozgatás, pallett_move,

- ESD környezet, esd_area,
- tiszta tér, cleanroom.

A tulajdonságok egy másik csoportja a humán szakértői tudásra vonatkozik, ezek az alábbiak:

- gépek száma, machine_number,
- mozgások hossza, movements,
- robotok száma, robot_numbers,
- megbízhatóság, reliability,
- teljesítmény, performance.

A szakértői jellemzők arra vonatkoznak, hogy milyen gyártórendszerben használták korábban a robotokat, és ott milyen tapasztalatokat gyűjtöttek. Ide írtam be a gépek számát, a robot mozgásainak hosszát, a tervezett robotok számát, a megbízhatóságot és a teljesítményt. Ez utóbbi a korábban ismertetett összefüggések között is szerepel, és azt mutatja meg, hogy a robot tényleges működési idejére, mennyi hasznos munkavégzési idő jut. Ez egy számított tapasztalati érték. A megbízhatóságot egy köztes jellemzőként értelmeztem, részben szakértői tapasztalat részben számolt értékek alapján.

A robotok tulajdonságait tartalmazó adatbázist feltöltöttem véletlenszerű adatokkal, amit egy python kóddal oldottam meg. A 21. táblázatban az adatbázis egy részlete látható.

21. táblázat Mobil robot adatbázis

	robot_id	manufacturer	type	loading_capacity	inside	outside	robot_weight	battery_capacity	time_battery_load	towing_ground_load	max_velocity	palletm_ove	esd_area	cleanroom	machine_numbers	movements	robot_numbers	reliability	performance
1	R0	['KUKA']	['250']	150.0	[True]	[False]	244.0	30.0	120.0	[False]	[False]	3.0	[False]	False	False	['high']	['long']	['extra_middle']	0.0,75.0
2	R1	['KUKA']	['Behemot']	500.0	[True]	[False]	283.0	30.0	60.0	[False]	[False]	5.0	[False]	False	False	['no_data']	['short']	['low']	85.0,4.0
3	R2	['Kivnon']	['100']	1000.0	[True]	[True]	426.0	1000.0	45.0	[True]	[False]	3.0	[False]	False	False	['high']	['very_short']	['middle']	90.0,49.0
4	R3	['ABB']	['300']	2000.0	[True]	[False]	375.0	120.0	120.0	[True]	[False]	5.0	[False]	False	False	['low']	['long']	['extra_middle']	0.0,84.0
5	R4	['Toyota']	['Multifunction']	10.0	[False]	[False]	353.0	360.0	120.0	[False]	[False]	4.0	[True]	False	False	['extra_high']	['no_data']	['middle']	0.0,45.0
6	R5	['EMS_group']	['250']	200.0	[False]	[True]	338.0	60.0	45.0	[False]	[False]	5.0	[False]	False	False	['high']	['extra_normal']	['huge']	85.0,70.0
7	R6	['Nipper']	['Scout']	20.0	[True]	[False]	148.0	30.0	120.0	[False]	[False]	8.0	[False]	False	False	['low']	['no_data']	['huge']	90.0,10.0
8	R7	['OMRON']	['Loader']	500.0	[True]	[False]	118.0	600.0	90.0	[False]	[False]	5.0	[True]	False	False	['extra_high']	['extra_normal']	['very_low']	0.0,99.0
9	R8	['Kivnon']	['250']	200.0	[True]	[False]	79.0	360.0	45.0	[False]	[False]	5.0	[False]	False	False	['medium']	['extra_long']	['very_low']	10.0,52.0
10	R9	['KUKA']	['Trasher']	5.0	[True]	[False]	320.0	480.0	60.0	[False]	[False]	4.0	[False]	False	False	['no_data']	['no_data']	['lot']	90.0,62.0
11	R10	['KUKA']	['200']	15.0	[True]	[False]	439.0	600.0	60.0	[False]	[True]	6.0	[False]	False	False	['high']	['long']	['very_low']	0.0,52.0
12	R11	['Toyota']	['Nexus']	5.0	[True]	[False]	249.0	90.0	120.0	[True]	[False]	8.0	[False]	False	False	['low']	['very_long']	['middle']	80.0,68.0
13	R12	['MIR']	['500']	2000.0	[True]	[False]	422.0	90.0	15.0	[False]	[False]	4.0	[False]	False	False	['extra_high']	['extra_normal']	['extra_middle']	100.0,57.0
14	R13	['OMRON']	['500']	1500.0	[False]	[False]	313.0	30.0	30.0	[False]	[False]	6.0	[False]	False	False	['extra_high']	['no_data']	['very_low']	30.0,40.0
15	R14	['ABB']	['Trasher']	50.0	[True]	[False]	403.0	45.0	120.0	[False]	[False]	15.0	[False]	False	False	['medium']	['extra_normal']	['two']	10.0,1.0
16	R15	['EMS_group']	['800']	5.0	[True]	[False]	377.0	600.0	60.0	[False]	[False]	5.0	[False]	False	False	['medium']	['extra_normal']	['huge']	0.0,88.0
17	R16	['KUKA']	['500']	50.0	[True]	[False]	238.0	90.0	120.0	[False]	[True]	6.0	[False]	False	False	['medium']	['very_short']	['huge']	30.0,18.0
18	R17	['KUKA']	['Phoenix']	300.0	[False]	[False]	76.0	180.0	30.0	[False]	[True]	15.0	[False]	False	False	['extra_high']	['very_long']	['single']	100.0,16.0
19	R18	['MIR']	['1000']	5.0	[True]	[False]	197.0	1000.0	30.0	[True]	[False]	8.0	[False]	False	False	['low']	['long']	['two']	85.0,24.0
20	R19	['Toyota']	['150']	10.0	[True]	[False]	164.0	120.0	120.0	[False]	[True]	6.0	[False]	False	False	['medium']	['normal']	['lot']	85.0,20.0
21	R20	['ABB']	['100']	5000.0	[False]	[False]	351.0	800.0	30.0	[False]	[False]	6.0	[False]	False	False	['extra_high']	['extra_long']	['single']	20.0,79.0
22	R21	['Bastian_solution']	['200']	5.0	[True]	[False]	301.0	60.0	90.0	[True]	[False]	4.0	[True]	False	False	['high']	['short']	['very_low']	80.0,19.0
23	R22	['MIR']	['Phoenix']	50.0	[True]	[False]	209.0	30.0	60.0	[True]	[False]	8.0	[False]	False	False	['no_data']	['normal']	['very_low']	50.0,15.0
24	R23	['KUKA']	['250']	1000.0	[False]	[False]	224.0	120.0	60.0	[False]	[False]	15.0	[False]	False	False	['extra_high']	['short']	['middle']	60.0,83.0
25	R24	['EMS_group']	['Nexus']	2500.0	[True]	[True]	402.0	800.0	30.0	[False]	[True]	6.0	[True]	False	False	['medium']	['extra_normal']	['extra_middle']	70.0,40.0
26	R25	['ABB']	['Trasher']	500.0	[True]	[False]	351.0	60.0	45.0	[True]	[False]	4.0	[True]	False	False	['no_data']	['extra_normal']	['lot']	70.0,67.0
27	R26	['EMS_group']	['Loader']	5.0	[True]	[False]	160.0	800.0	120.0	[True]	[False]	5.0	[False]	False	False	['medium']	['extra_long']	['extra_middle']	85.0,1.0
28	R27	['Kivnon']	['Tower']	5000.0	[True]	[False]	455.0	180.0	60.0	[False]	[False]	8.0	[False]	False	False	['extra_high']	['extra_long']	['extra_middle']	50.0,74.0
29	R28	['ABB']	['100']	1000.0	[True]	[False]	76.0	1000.0	90.0	[False]	[False]	4.0	[False]	False	False	['low']	['no_data']	['two']	50.0,29.0
30	R29	['EMS_group']	['Behemot']	5.0	[True]	[False]	221.0	45.0	30.0	[False]	[False]	4.0	[False]	False	False	['high']	['very_long']	['very_low']	40.0,51.0
31	R30	['KUKA']	['Trasher']	2000.0	[True]	[False]	313.0	90.0	90.0	[True]	[False]	4.0	[False]	False	False	['medium']	['very_short']	['low']	60.0,8.0
32	R31	['OMRON']	['Tower']	150.0	[True]	[False]	287.0	45.0	120.0	[True]	[False]	3.0	[False]	False	False	['low']	['normal']	['lot']	50.0,43.0
33	R32	['Nipper']	['1000']	150.0	[False]	[True]	241.0	800.0	30.0	[False]	[False]	8.0	[False]	False	False	['high']	['very_long']	['single']	85.0,61.0
34	R33	['ABB']	['Loader']	2000.0	[True]	[False]	404.0	90.0	90.0	[False]	[False]	6.0	[False]	False	False	['medium']	['extra_normal']	['two']	0.0,30.0
35	R34	['MIR']	['150']	100.0	[True]	[False]	200.0	120.0	45.0	[False]	[False]	15.0	[False]	False	False	['low']	['very_short']	['very_low']	90.0,42.0
36	R35	['KUKA']	['800']	1000.0	[False]	[False]	329.0	45.0	120.0	[False]	[False]	5.0	[False]	False	False	['high']	['extra_long']	['extra_middle']	50.0,87.0

Ezek után írtam egy python kódot, ami a fenti táblázatból kiolvassa, majd a CLIPS számára „tényekként” feldolgozható formába hozza az adatokat, így megkaptam a tényeket tartalmazó CLIPS fájlt, amelyből egy részlet a mellékletben, a 119. oldalon megtalálható.

Ezek után a szabályok létrehozása következett, amelyeket úgy fogalmaztam meg, hogy a szakértői adatokat alapul véve válassza ki a robotot a tények alapján. Ezen kívül bármilyen más lehetőség is elképzelhető, a korábban felvetett ötleteket is alapul véve. Ki lehet indulni a robotból, vagy a gyártórendszerből is.

A teszt során a felhasználónak három kérdést teszt fel a program, amelyek a gyártórendszerre és a robot teherbírására vonatkoznak:

- Hány gép van a gyártórendszerben?
- Milyen sokat kell mozognia a robotnak?
- Mekkora a szükséges teherbírása?

Az első két kérdés szándékosan nem konkrét számértéket kér, hanem a szakértői intuíciót és tapasztalatot felhasználva lehet őket megválaszolni. A 22. táblázatban láthatóak a tulajdonságok és azok lehetséges értékei.

22. táblázat Robot tulajdonságok szakértői tudással kiegészítve

Csoport	Tulajdonság magyarul	Tulajdonság értékei
Robot tulajdonság adatlapról	gyártó	Manufacturer = ['MIR', 'OMRON', 'KUKA', 'ABB', 'Toyota', 'Bastian_solution', 'EMS_group', 'Kivnon', 'Nipper']
	típus	Type = [100, 150, 200, 250, 300, 500, 800, 1000, 'Scout', 'Adventurer', 'Loader', 'Tower', 'Nexus', 'Phoenix', 'Trasher', 'Standard', 'Multifunction', 'Behemot']
	beltéri	Inside = [True, True, True, True, False] #80%
	kültéri	Outside = [True, False, False, False, False] #20%
	terhelhetőség	Loading_Capacity = [5, 10, 15, 20, 50, 100, 150, 200, 300, 500, 800, 1000, 1500, 2000, 2500, 5000] #[kg]
	akkumulátor kapacitás	Battery_capacity = [30, 45, 60, 90, 120, 180, 360, 480, 600, 800, 1000] #[minutes]
	töltési idő	Battery_load_time = [15, 30, 45, 60, 90, 120]
	kocsi húzás	Towing = [False, False, False, False, True]
	földről felvétel	Ground_load = [False, False, False, False, True]
	raklap mozgatás	Pallett_move = [False, False, False, False, True]
	maximális sebesség	Max_velocity = [3, 4, 5, 6, 8, 15] #[km/h]
Szakértői értékek	gépek száma	Machine_number = ['low', 'medium', 'high', 'extra_high', 'no_data']
	mozgások	Movements = ['very_short', 'short', 'normal', 'extra_normal', 'long', 'very_long', 'extra_long', 'no_data']
	robotok száma	Robot_numbers = ['single', 'two', 'very_low', 'low', 'middle', 'extra_middle', 'lot', 'huge', 'no_data']
Működési értékek	megbízhatóság	Reliability = [0,10,20,30,40,50,60,70,80,85,90,95,100]
	teljesítmény	Performance

A fejlesztés ezen fázisában a rendszer még „merev”, amit úgy értek, hogy csak egy-egy válasz lehetőséget fogad el, és ahhoz keresi a robotot a tényhalmazból. Vagyis az első kérdésre a „kevés-közepes-sok-nagyon sok” ill. a „nincs adat” lehetőségekből lehet választani. Ezt azért alakítottam ki így mert egy emberi szakértőnek vannak fogalmai egy-egy gyártórendszer nagyságáról, de nem biztos, hogy vannak konkrét számai az egyes gyártórendszerekhez tartozó gépekről. Valamint a gyártórendszer még lehet tervezés alatt is, esetleg egy-egy gép vagy gyártósor határát nem is olyan egyszerű meghatározni, ezért így (bár

nem tűnik annyira pontosnak) mégis jóval pontosabban írja le a gyártórendszer felépítését, mint egy, ebben a fázisban még talán nem is ismert egzakt szám.

A kültéri/beltéri megoszlásnál a beltéri arányát 80%-ra választottam, mert ez a jellemzőbb. Viszont elképzelhető olyan generált eset, amikor mindkét változó értéke „hamis” lesz, ami valós esetben nem lehetséges, így a programba ide tettem egy feltételt, és amennyiben a véletlenszerű generálásnál egyik sem „igaz” akkor a robot a beltéri besorolást kapja.

Az adatbázist feltöltöttem egy minta robottal és 100 véletlenszerűen generált paramétereket tartalmazó robottal, és a három kérdés alapján a szabályokat implementáltam a CLIPS rendszerbe, így már volt egy működő képes szakértői rendszer prototípusom az egész elképzelésről. A szabályok alapján történik a kiválasztás, így a tények halmazából a kritériumoknak megfelelő robotok kerülnek kiválasztásra.

Egy teszt futás eredménye az alábbi:

```
Milyen machine_number ertek kell? ('low', 'medium', 'high', 'extra_high', 'no_data') low
Milyen movements ertek kell? ('very_short', 'short', 'normal', 'extra_normal', 'long', 'very_long',
'extra_long', 'no_data') short
Minimum loading_capacity (kg)? 40
*** Javasolt robot: R90 Kivnon 0
*** Javasolt robot: R87 KUKA 0
*** Javasolt robot: R34 MIR 5
CLIPS>
```

A program a beírt feltételek alapján leszűri a tényeket, és kiírja az eredményt. Az első „R90” a robot sorszámára utal a táblázatban. Ezek alapján a teszt eredmények alapján három robot felel meg a kritériumoknak. Ezen három robot közül hogyan érdemes választani? Erre egy többcélú optimalizációs megoldást javaslok.

Készítettem egy többcélú optimalizációt a VIKOR módszerrel, az alábbi bemenő adatokkal, (nem a fenti három robotra, hanem csak példának):

- Terhelhetőség, loading_capacity – haszon jellegű
- Akkumulátor kapacitás, battery capacity – haszon jellegű
- Akkumulátor töltési idő, battery time load – költség jellegű
- Maximális sebesség, max velocity – haszon jellegű
- Megbízhatóság, reliability – haszon jellegű
- Teljesítmény, performance – haszon jellegű

A 23. táblázatban foglaltam össze az adatokat, az első 5 robotra. Az első sorban az adott tulajdonság súlya (fontossága) szerepel, a korábbi (33) és (34) - es képletekben „w”-vel volt jelölve.

23. táblázat Robot tulajdonságok táblázat VIKOR-hoz

	Terhelhetőség	Akkumulátor kapacitás	Töltési idő	Maximális sebesség	Megbízhatóság	Teljesítmény
súly	0,2	0,2	0,1	0,2	0,1	0,2
R0	150	30	120	3	0	75
R1	500	30	60	5	85	4
R2	1000	1000	45	3	90	49
R3	2000	120	120	5	0	84
R4	10	360	120	4	0	45

Kiválasztottam a minimális és maximális értékeket:

24. táblázat Minimális és maximális értékek

min	10	30	45	3	0	4
max	2000	1000	120	5	90	84

Meghatároztam az „f_{ij}”-ket a haszon és a költség jellegű jellemzőkre:

25. táblázat Az "Fij" értékek

haszon						
f _{ij}						
R0	0,07035176	0		0	0	0,8875
R1	0,24623116	0		1	0,944444	0
R2	0,49748744	1		0	1	0,5625
R3	1	0,092783505		1	0	1
R4	0	0,340206186		0,5	0	0,5125
költség						
f _{ij}						
R0			0			
R1			0,8			
R2			1			
R3			0			
R4			0			

Majd az „ S_i ” és „ R_i ” számításához az zárójelben lévő segédváltozót a (33) képletből, és az eredményeket beírtam a 26. táblázatba.

$$S_i = \sum_{j=1}^n w_j * (f_j^* - f_{ij}) \quad (33)$$

26. táblázat Segédváltozók értékei

R0	0,18592965	0,2	0,1	0,2	0,1	0,0225
R1	0,15075377	0,018556701	0,02	0	0,005556	0,2
R2	0,10050251	-0,181443299	0	0,2	0	0,0875
R3	0	0	0,1	0	0,1	0
R4	0,2	-0,049484536	0,1	0,1	0,1	0,0975

Végül kiszámoltattam az értékeket, amelyek a 27. táblázatban vannak.

27. táblázat R_i , S_i és Q_i értékek

	R_i	S_i				Q_i
R0	0,2	0,808429648		v	0,5	1
R1	0,2	0,394866025		S*	0,2	0,660138502
R2	0,2	0,206559214		S-	0,80843	0,505390281
R3	0,1	0,2		R*	0,1	0
R4	0,2	0,548015464		R-	0,2	0,785994827

Ezek alapján a legkisebb Q értékű robot lenne az optimális a tulajdonságok alapján. Ez jelen esetben az R3-as robot. Viszont visszanézve a 23. táblázatot, aminek egy részlete 28. táblázatban is látható, feltűnik, hogy ennek a robotnak arra a gyártórendszerre nézve, amiben tesztelték a megbízhatósága nulla. Ezt egy későbbi verzióban már előre ki is lehetne szűrni. Ha ezt figyelembe vesszük a döntésünknél, akkor célszerű lenne a második legjobbat választani, ami pedig az R2-es robot. Ennek adatai valóban jobban festenek az R3-nál.

28. táblázat R2 és R3 robotok összevetése

	Terhelhetőség	Akkumulátor kapacitás	Töltési idő	Maximális sebesség	Megbízhatóság	Teljesítmény
R2	1000	1000	45	3	90	49
R3	2000	120	120	5	0	84

Látható, hogy az R3-as robot a terheltségével, sebességével és teljesítményével emelkedett ki a mezőnyből. Ugyanakkor az R2-es robot hatalmas akkumulátor kapacitása, rövid töltési ideje, magas megbízhatósága

bőven ellensúlyozhatja az alacsony sebességét és szerény teljesítményét. Az R3-as robot nulla megbízhatósága és viszonylag magas teljesítménye látszólag egy gyakorlati ellentmondás, ugyanakkor vegyük figyelembe, hogy a 21. táblázatban generált adatok jelennek meg.

Végül az egész mobil robot adatbázisra elkészítettem a VIKOR módszert Excel-ben, a fenti minta alapján, de a 29. táblázatban szereplő tulajdonságokkal és súlyokkal.

29. táblázat VIKOR módszer súlyok a robot tulajdonságokra

	Terhelhetőség	Robot tömege	Akkumulátor kapacitás	Akkumulátor töltési ideje	Maximális sebesség	Megbízhatóság	Teljesítmény
súly	0,1	0,05	0,15	0,1	0,2	0,2	0,2

Az első 20 robot sorrendje:

30. táblázat VIKOR módszerrel a robotok sorrendje

Helyezés	Robot száma	Qi értéke
1.	R2	0,118492
2.	R60	0,126879
3.	R73	0,133719
4.	R95	0,153235
5.	R83	0,172604
6.	R55	0,176621
7.	R67	0,215932
8.	R12	0,233017
9.	R85	0,239724
10.	R62	0,243729
11.	R87	0,249704
12.	R15	0,254533
13.	R91	0,255645
14.	R78	0,257896
15.	R52	0,258463
16.	R75	0,263446
17.	R96	0,266588
18.	R9	0,268287
19.	R59	0,272423
20.	R7	0,274902

A 30. táblázatban látható, hogy az R2-es robot a teljes 101 db robotot tartalmazó adatbázisból is az első helyen végzett. Itt fontos azt megjegyezni, hogy a VIKOR módszer sajátosságai miatt a két esetben (amikor a 4 robotot ill. amikor a 101 robotot hasonlítottam össze) mások voltak a minimális és maximális értékek az egyes mezőkben, valamint az egyes súlyozások is a teljes 102 soros táblázatra is, így mások lettek az S_i , R_i és Q_i számértékek is, és ez az oka, hogy amikor 4 robotot hasonlítottam össze akkor nem az R2 végzett az első helyen.

Ez az esettanulmány igazolja a módszer működő képességét. Összegezve az elvégzett kutatási lépések az alábbiak voltak:

1. Adatok előállítása python-al „csv” fájlba. Összesen 101 robot szerepelt az adatbázisban.
2. Adatok kiolvasása a „csv” fájlból egy másik python programmal, és átírva a CLIPS – ben feldolgozható tényekké egy „clp” fájlba.
3. Szabályok és CLIPS kód létrehozása és futtatása CLIPS – ben.
4. CLIPS – ben az eredmények összegzése kimentése Excel-be.
5. Az eredmények értékelése, a robotok összevetése VIKOR módszerrel.
6. Az optimálisnak ítélt robot adatainak szakértői elemzése és a kiválasztás megerősítése, vagy felülbírlata.

Az így összeállított rendszeremet nagyszámú robotra (101 db) és nagyszámú tulajdonságra (19 tulajdonság) is teszteltem. A módszer lényege tehát, hogy egyfajta előszűrést végez a szakértői rendszer a robot vagy gyártórendszer tulajdonságok alapján, majd egy multikritériumos összehasonlító módszerrel sorrendet határoz meg a potenciálisan számításba vehető robotok között, ezzel elvégezve a kutatás céljából kitűzött döntéstámogatást. Végül az első három dobogós választott robot tulajdonságait mérlegelve hozhatja meg a döntéshozó a felelős döntést.

Több változót is szándékosan elhagytam, amiket figyelembe kellene venni, pl.: tud-e vontatni a robot, vagy tud-e raklapot mozgatni. Az ezekre való szűrést a korábbi esettanulmányok miatt már túl egyértelműnek ítélttem meg, és inkább más paramétereket állítottam a fókuszba, hogy a módszer és a felépített prototípus rendszer erősségei jobban látszódnak. Amit itt még fontos kiemelni, hogy ez a teszt a költség változót figyelmen kívül hagyja, így a

gazdasági szempontokat nem a közvetlen ismert költségek oldaláról ragadja meg, hanem áttételesen a megbízhatóságon keresztül az üzemeltetési költségek felől. A módszer működőképességét ez nem befolyásolja, csak nem akartam az összes költség tényezőre kihegyezni a tesztet, mert az árak nem elérhetőek a katalógus adatokból, de maga a 21. táblázat és a módszer is alkalmas arra, hogy ezeket gyorsan integráljuk egy 20. tulajdonságnak.

A teszt eredmények igazolják, hogy a módszertan két keretrendszerben elkészült teszteléssel láthatóan működőképes.

3.8 Újdonságtartalom és hozzájárulás

A szakirodalmi áttekintés részben ismerttetett módszerekhez képest az én kutatásomat az teszi egyedivé, hogy a tudásbázist a szakértői rendszerrel és a VIKOR módszerrel még nem kombinálták robot kiválasztásra. Újdonság továbbá a skálázhatóság, ami 101 robotot és 19 jellemzőt fel le, és algoritmikusan bővíthető. A rendszeremben nem csak objektív vagy szubjektív kritériumokat használok, hanem logikai rendszerrel strukturálom a szakértői tudást. A szakirodalmi eredmények és az én kutatások összevetése a 31. táblázatban látható összefoglalóan.

31. táblázat A kutatás eredményeinek összevetése

Kutatási fókusz	Publikációkban	Az én kutatásom
MCDM módszerek (fuzzy VIKOR, TOPSIS...)	Széles spektrum — fuzzy VIKOR, TOPSIS, AHP, CRITIC stb.	VIKOR alkalmazása konkrétan, de ötvözve tudásbázissal és szakértői rendszerrel
Fuzzy DSS rendszerek	Fuzzy DSS javaslatok ipari robotválasztásra	Intelligens rendszer, adat- és tudásbázissal, logikai reprezentációval
Skálázhatóság, nagy robotállomány	Ritkán említve a szakirodalomban	Magas — több mint 100 robot és sok tulajdonság skálázhatóan kezelhető

A harmadik tézis ismertetése:

Kidolgoztam egy szakértői rendszeren alapuló döntéstámogató keretrendszert ipari mobil robotok gyártórendszerbe történő integrálására. A módszer elsőként egyesíti a robotok adatlapjából származó formális paramétereiket, a humán szakértők gyakorlati tudását és a korábbi

gyártórendszerekből származó működési jellemzőket. A módszer újdonsága, hogy a mobil robot kiválasztása nem kizárólag technikai specifikációkra, hanem tapasztalati tudásra és a gyártási környezet szakértői ismeretére is épül.

A harmadik tézis első alpontja:

Létrehoztam egy tudásbázis-építési módszertant, amely logikai reprezentáció (ítéletlogika, fuzzy megközelítés, CLIPS keretrendszer) segítségével teszi lehetővé a mobil robotok jellemzőinek és a gyártórendszerek igényeinek összehangolását. Az eljárás bizonyítottan skálázható: nagyszámú (100 feletti) robot és több tucat jellemző együttes kezelésére is alkalmas, miközben a szakértői tudást integrált formában képes reprezentálni és feldolgozni.

A harmadik tézis második alpontja:

A szakértői rendszer döntési eredményeit többcélú optimalizációval (VIKOR módszer) egészítettem ki, amely rangsorolást biztosít az előszűrés során kiválasztott alternatívák között. A kutatás igazolta, hogy a kombinált módszer képes a mobil robotok közül objektív és megalapozott sorrendet kialakítani, miközben lehetőséget ad a szakértői korrekcióra is. Ezáltal a rendszer a gyártórendszerekben történő robotintegrációhoz megbízható és gyakorlati döntéstámogatást nyújt.

4 Jövőbeni tervek

Az ipari mobil robotok várható további terjedése miatt a kutatás folytatásának szükségessége nem szorul külön magyarázatra. Az dolgozatban ismertetett módszerek alkalmazhatóságát a bemutatott esettanulmányok alátámasztják.

A jövőbeni fejlesztési lehetőségek közül a saját szimulációhoz kapcsolódóan várható nagyobb előrelépés, hiszen akkor lehetőség lesz egy komplett gyártási folyamat szimulálására. Így a mobil robotok funkcionalitása is tesztelhetővé válik, valamint a valóságban meglévő különbségeket a szimulációban is be lehet jeleníteni, ezzel visszacsatolva a szakértői rendszer adatbázisára és tudásbázisára.

A tudásbázis alapú szakértői rendszert is érdemes fejleszteni, a 3. fejezetben leírtakra alapozva, kiegészítve még több jellemzővel, valós adatokkal, és így felépíteni a szakértői rendszer tudásbázisát. Az így módosított szakértői rendszer egy újabb verziója már tesztelhető lehet ipari környezetben is, és ezekkel a tapasztalatokkal kiegészítve a fejlesztés irányát is pontosítani lehet.

Összefoglalás

Disszertáciomban az ipari mobil robotok kiválasztási folyamatával kapcsolatban végeztem kutatásokat. A mobil robotokat használó gyártórendszer modelleket egyszerűsítettem, és felírtam egy új gyártórendszer modellt, melyben a mobil robotnak központi szerepet adtam, így alkottam meg a „mobil robot központú” gyártórendszer modellt. A modellt felhasználva vizsgáltam több szállítási feladatot, valamint egy genetikus algoritmus kiértékelő függvényeként is felhasználtam.

A kutatás másik részében egy szakértői rendszer prototípusát készítettem el, amivel a mobil robotok adatbázisából készített tudásbázist teszteltem. A tudásbázis a tények és a szabályok halmaza jól skálázható mind a robotok számában, mind a robot tulajdonságainak számában, ahogyan a 101 roboton és 19 tulajdonságon elvégzett tesztelés ezt bizonyította. Az objektív, csak a robotokra vonatkozó tudást kiegészítettem szakértői tapasztalattal és a gyártórendszer jellemzőivel, amelyeket logikai formában tettem kezelhetővé a szakértői rendszerben.

A kutatásom újdonsága abban áll, hogy elsőként dolgoztam ki szakértői rendszeren alapuló, skálázható döntéstámogató módszert ipari mobil robotok gyártórendszerbe történő integrálására. A megoldás egyesíti az adatlapokon szereplő műszaki paramétereket, a humán szakértői tapasztalatot és a gyártási környezet komplex követelményeit, majd ezekre építve többcélú optimalizációval megalapozott rangsorolást ad. A rendszer így átfogóan és gyakorlati módon támogatja az ipari döntéshozókat a megfelelő mobil robot kiválasztásában.

Az elvégzett kutatások alapján a mobil robotok gyártórendszerbe illesztésének döntéstámogatása az új sorokat tervező mérnököknek jelent segítséget.

Új tudományos eredmények: a doktori értekezése tézisei

1. Tézis:

Alkottam egy új modellezési módszert, amely a mobil robotot a gyártórendszerrel leválasztva képes kezelni. Ez a modell egy 10 állapottal és 8 átmenettel felrajzolt Petri-háló, amelyben a mobil robot szempontjából a gyártórendszer szükséges részei vannak megragadva, amelyek a következők: honnan, mit (fizikai méretek, súly, darabszám) és hova kell szállítani, milyen útvonalon, mekkora távolságra, hogyan tudja felvenni és lerakni a robot, és mekkora az akkumulátor töltés csökkenése a szállítási feladat hatására. A Petri-hálós modellt az akkumulátor kapacitás teszi korlátossá, minden állapot elérhető, és holtponmentes. Az új módszer neve a mobil robot központú gyártórendszer modell.

2. Tézis:

A 2. tézis „A” része: A mobil robot központú gyártórendszer szemléletét felhasználva igazoltam, hogy egy egyirányú, két pont közötti szállítási feladatra a robot akkumulátor kapacitása két nagyságrendet is átölelő elméleti határok között vizsgálható, és az ideális méret a feladat függvényében erőforráshatékonyan meghatározható, tehát a módszer jól skálázható.

A 2. tézis „B” része: Igazoltam, hogy a mobil robot központú gyártórendszer modellel felírható olyan számítási modell, amely magasabb szintű optimálási algoritmusok kiértékelő függvényeként használható. Az elvégzett esettanulmányban a genetikus algoritmus egyedeit kiértékelő fitness függvényként való felhasználás a valós gyártási folyamatokat veszi figyelembe és így ad jellemző paramétert, ami emiatt a kísérletek alapján 10 generáció alatt elérte az optimális értéket.

Az 1. és 2. tézist az alábbi elfogadott, publikálás alatt lévő Q1-es publikáció támasztja alá:

M. Boleraczi, I. G. Gyurika: Data-Centric Engineering: Industrial mobile robot-based manufacturing system modelling potential

3. Tézis:

Kidolgoztam egy szakértői rendszeren alapuló döntéstámogató keretrendszert ipari mobil robotok gyártórendszerbe történő integrálására. A módszer elsőként egyesíti a robotok adatlapjából származó formális paramétereiket, a humán szakértők gyakorlati tudását és a korábbi gyártórendszerekből származó működési jellemzőket. A módszer újdonsága, hogy a mobil robot kiválasztása nem kizárólag technikai specifikációkra, hanem tapasztalati tudásra és a gyártási környezet szakértői ismeretére is épül.

3.1. A harmadik tézis első alpontja:

Létrehoztam egy tudásbázis-építési módszertant, amely logikai reprezentáció (ítéletlogika, fuzzy megközelítés, CLIPS keretrendszer) segítségével teszi lehetővé a mobil robotok jellemzőinek és a gyártórendszerek igényeinek összehangolását. Az eljárás bizonyítottan skálázható: nagyszámú (100 feletti) robot és több tucat jellemző együttes kezelésére is alkalmas, miközben a szakértői tudást integrált formában képes reprezentálni és feldolgozni.

3.2. A harmadik tézis második alpontja:

A szakértői rendszer döntési eredményeit többcélú optimalizációval (VIKOR módszer) egészítettem ki, amely rangsorolást biztosít az előszűrés során kiválasztott alternatívák között. A kutatás igazolta, hogy a kombinált módszer képes a mobil robotok közül objektív és megalapozott sorrendet kialakítani, miközben lehetőséget ad a szakértői korrekcióra is. Ezáltal a rendszer a gyártórendszerekben történő robotintegrációhoz megbízható és gyakorlati döntéstámogatást nyújt.

A 3. tézist az alábbi publikáció támasztja alá:

- [83] M. Boleraczki, „Development of a knowledge base for an expert system supporting the decision of industrial mobile robot selection”, J. Phys.: Conf. Ser., köt. 2714, sz. 1, o. 012004, febr. 2024, doi: 10.1088/1742-6596/2714/1/012004.

New scientific results: thesis statement of the doctoral dissertation

1. Thesis: I have developed a new modeling method that can manage the mobile robot independently of the production system. This model is a Petri net with 10 states and 8 transitions, which represents the necessary parts of the production system from the perspective of the mobile robot, as follows: where to pick up what (physical dimensions, weight, number of pieces) and where to deliver it, what route to take, which distance to travel, how the robot can pick up and put down the load, and how much the battery charge will decrease as a result of the transport task. The Petri net model is limited by battery capacity, all states are accessible, and there are no dead locks. The new method is called the mobile robot-centric manufacturing system model.

2. Thesis:

Part A of Thesis 2: Using the mobile robot-centric manufacturing system concept, I demonstrated that for a one-way transport task between two stations, the robot's battery capacity can be analyzed within theoretical limits covering two orders of magnitude, and the ideal size can be determined in a resource-efficient manner depending on the task, meaning that the method is highly scalable.

Part B of Thesis 2: I demonstrated that a computational model can be written using the mobile robot-centric manufacturing system model, which can be used as an evaluation function for higher-level optimization algorithms. In the case study, using genetic algorithms as a fitness function to evaluate individuals takes real manufacturing processes into account and thus provides a characteristic parameter, which therefore reached its optimal value in 10 generations based on the experiments.

Theses 1 and 2 are supported by the following accepted Q1 publication, currently in press:

M. Boleraczi, I. G. Gyurika: Data-Centric Engineering: Industrial mobile robot-based manufacturing system modelling potential

3. Thesis: I have developed a decision support framework based on an expert system for integrating industrial mobile robots into manufacturing systems. The method combines formal parameters from robot data sheets, the practical knowledge of human experts, and operational characteristics from previous manufacturing systems.

The novelty of the method is that the selection of mobile robots is based not only on technical specifications, but also on empirical knowledge and expert knowledge of the manufacturing environment. The novelty of the method is that the selection of mobile robots is based not only on technical specifications, but also on empirical knowledge and expert knowledge of the manufacturing environment.

3.1. First sub thesis of the third thesis:

I have created a knowledge base construction methodology that uses logical representation (propositional logic, fuzzy approach, CLIPS framework) to enable the alignment of mobile robot characteristics and manufacturing system requirements. The method is proven to be scalable: it is suitable for handling a large number (over 100) of robots and dozens of attributes, while being able to represent and process expert knowledge in an integrated form.

3.2. Second sub thesis of the third thesis: I have combined the decision results of the expert system with multi-objective optimization (VIKOR method), which provides a ranking among the alternatives selected during the preliminary ranking. The research confirmed that the combined method is capable of establishing an objective and feasible ranking among mobile robots, while also allowing for expert correction. Thus, the system provides reliable and practical decision support for robot integration in manufacturing systems.

Thesis 3 is supported by the following publication:

[83] M. Boleraczki, „Development of a knowledge base for an expert system supporting the decision of industrial mobile robot selection”, J. Phys.: Conf. Ser., köt. 2714, sz. 1, o. 012004, febr. 2024, doi: 10.1088/1742-6596/2714/1/012004.

Ábrajegyzék

1. ábra Gyártórendszer tervezési folyamat	13
2. ábra Gyártási idők értelmezése.....	22
3. ábra Robot ágens sematikus képe.....	28
4. ábra 10 kihívás a robotika területén	33
5. ábra A CLIPS szakértői keretrendszer képe	34
6. ábra Hagyományos szállítószalagos gyártórendszer felépítése.....	54
7. ábra Rugalmas gyártórendszer sematikus rajza	55
8. ábra Valós FMS rendszer képe.....	55
9. ábra FMS gyártórendszer Petri-hálója	56
10. ábra AGV-vel kiegészített gyártórendszer.....	57
11. ábra Automatikus munkadarab mozgató modellje.....	58
12. ábra Gyártórendszer modell mobil robottal	59
13. ábra A mobil robot modellje a gyártórendszerhez kapcsolva	59
14. ábra Mobil robot és gyártórendszer kapcsolat Petri-hálós alap modellje ...	61
15. ábra Alap modell Petri-háló analízis eredmények	62
16. ábra A mobil robot központú gyártórendszer modell áttekintő ábrája.....	67
17. ábra Gyártórendszer részlet vázlata háromféle termék útvonallal.....	69
18. ábra A mobil robot modell állapotai	72
19. ábra Kétpontos szállítási feladat szimulációja	73
20. ábra A futtatott teszt eredménye szövegesen	73
21. ábra Az akkumulátor kapacitás és a csomagszámok összefüggése.....	74
22. ábra Logaritmikus skálán ábrázolva az akku kapacitás	75
23. ábra Állapotgép többféle termék gyártására.....	77
24. ábra Termékek gyártásának grafikonja a darabszám, a távolságok és a töltések számának összefüggésében	78
25. ábra Fitness függvény értékek	80
26. ábra Néhány piacon elérhető mobil robot.....	84
27. ábra A szakértői rendszer munkafolyamati ábrája	86

Táblázatok jegyzéke

1. táblázat MCDM módszerek összehasonlítása	19
2. táblázat VIKOR módszer példa adatok, és számolás	45
3. táblázat Si és Ri értékek számítása	45
4. táblázat Segédértékek Qi számításához	45
5. táblázat A Qi számított értékei	46
6. táblázat TOPSIS módszerhez paraméterek	47
7. táblázat Kritériumok súlyozása.....	47
8. táblázat Normalizált értékek a „min” jobb esetben	48
9. táblázat Normalizált értékek a "max" jobb esetben	48
10. táblázat Normalizált és súlyozott értékek	49
11. táblázat Szélsőértékű megoldások.....	49
12. táblázat Súlyozott szélsőértékek	50
13. táblázat A D+ és D- értékei	50
14. táblázat A C* számítása	50
15. táblázat Alternatíva eredmények.....	51
16. táblázat Genetikus algoritmus eredmények I.	79
17. táblázat Genetikus algoritmus eredmények II.	80
18. táblázat A mobil robotok adatbázisa	86
19. táblázat A tudásbázis részlete.....	88
20. táblázat Az adatbázis egy részlete	90
21. táblázat Mobil robot adatbázis.....	94
22. táblázat Robot tulajdonságok szakértői tudással kiegészítve.....	95
23. táblázat Robot tulajdonságok táblázat VIKOR-hoz	97
24. táblázat Minimális és maximális értékek	97
25. táblázat Az "Fij" értékek	97
26. táblázat Segédváltozók értékei	98
27. táblázat Ri, Si és Qi értékek.....	98
28. táblázat R2 és R3 robotok összevetése	98
29. táblázat VIKOR módszer súlyok a robot tulajdonságokra	99
30. táblázat VIKOR módszerrel a robotok sorrendje	99
31. táblázat A kutatás eredményeinek összevetése	101

Irodalomjegyzék

- [1] G. Seliger, Proceedings / 11th Global Conference on Sustainable Manufacturing innovative solutions ; Berlin, Germany, 23rd - 25th September, 2013 ; proceedings. Berlin: Univ.-Verl. der TU, 2013.
- [2] Sz. Kovács, „Ipar 4.0 koncepció és kulcselemei”, International Journal of Engineering and Management Sciences (IJEMS), köt. 7., doi: 10.21791/IJEMS.2022.2.8.
- [3] E. Oztemel és S. Gursev, „Literature review of Industry 4.0 and related technologies”, J Intell Manuf, köt. 31, sz. 1, o. 127–182, jan. 2020, doi: 10.1007/s10845-018-1433-8.
- [4] M. C. Zizic, M. Mladineo, N. Gjeldum, és L. Celent, „From Industry 4.0 towards Industry 5.0: A Review and Analysis of Paradigm Shift for the People, Organization and Technology”, Energies, köt. 15, sz. 14, o. 5221, júl. 2022, doi: 10.3390/en15145221.
- [5] J. Alves, T. M. Lima, és P. D. Gaspar, „Is Industry 5.0 a Human-Centred Approach? A Systematic Review”, Processes, köt. 11, sz. 1, o. 193, jan. 2023, doi: 10.3390/pr11010193.
- [6] S. Wang, J. Wan, D. Zhang, D. Li, és C. Zhang, „Towards smart factory for industry 4.0: a self-organized multi-agent system with big data based feedback and coordination”, Computer Networks, köt. 101, o. 158–168, jún. 2016, doi: 10.1016/j.comnet.2015.12.017.
- [7] S. B. Gershwin, „The future of manufacturing systems engineering”, International Journal of Production Research, köt. 56, sz. 1–2, o. 224–237, jan. 2018, doi: 10.1080/00207543.2017.1395491.
- [8] W. R. Nyemba és C. Mbohwa, „Modelling, Simulation and Optimization of the Materials Flow of a Multi-product Assembling Plant”, Procedia Manufacturing, köt. 8, o. 59–66, 2017, doi: 10.1016/j.promfg.2017.02.007.
- [9] J. Wang, Q. Chang, G. Xiao, N. Wang, és S. Li, „Data driven production modeling and simulation of complex automobile general assembly plant”, Computers in Industry, köt. 62, sz. 7, o. 765–775, szept. 2011, doi: 10.1016/j.compind.2011.05.004.
- [10] L. Nyeki, „A Petri hálók alkalmazási lehetőségei az oktatási folyamat modellezésében”, Conference: X. Trefort Ágoston Szakképzés- és Felsőoktatás-pedagógiai Konferencia, 2020.
- [11] N. Chungoora és mtsai., „Extending product lifecycle management for manufacturing knowledge sharing”, Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, köt. 226, sz. 12, o. 2047–2063, dec. 2012, doi: 10.1177/0954405412461741.
- [12] D. Mourtzis, „Simulation in the design and operation of manufacturing systems: state of the art and new trends”, International Journal of Production Research, köt. 58, sz. 7, o. 1927–1949, ápr. 2020, doi: 10.1080/00207543.2019.1636321.
- [13] I. Um, H. Cheon, és H. Lee, „The simulation design and analysis of a Flexible Manufacturing System with Automated Guided Vehicle System”,

Journal of Manufacturing Systems, köt. 28, sz. 4, o. 115–122, dec. 2009, doi: 10.1016/j.jmsy.2010.06.001.

- [14] I. Németh, J. Püspöki, A. B. Viharos, L. Zsóka, és B. Pirka, „Layout configuration, maintenance planning and simulation of AGV based robotic assembly systems”, IFAC-PapersOnLine, köt. 52, sz. 13, o. 1626–1631, 2019, doi: 10.1016/j.ifacol.2019.11.433.
- [15] G. Fedorko, V. Molnár, J. Strohmandl, P. Horváthová, D. Strnad, és V. Cech, „Research on Using the Tecnomatix Plant Simulation for Simulation and Visualization of Traffic Processes at the Traffic Node”, Applied Sciences, köt. 12, sz. 23, o. 12131, nov. 2022, doi: 10.3390/app122312131.
- [16] J. Kopec, L. Lachvajderová, M. Kliment, és P. Trebuňa, „SIMULATION PROCESSES IN COMPANIES USING PLM AND TECNOMATIX PLANT SIMULATION SOFTWARE”, AS, köt. 7, sz. 3, o. 13–18, szept. 2021, doi: 10.22306/asim.v7i3.61.
- [17] J. H. D. C. Q. Gonzalez és N. Kuwahara, „Production Line Virtualization Process Using Plant Simulation Tool”, Int J Innov Educ Res, köt. 9, sz. 9, o. 188–201, szept. 2021, doi: 10.31686/ijer.vol9.iss9.3329.
- [18] L. N. Chilukuri, V. K. Sriramula, S. Veerappan, és M. Shankar, „Implementation of Simulation Practices for Plant Optimization”, in Industry 4.0 and Advanced Manufacturing, A. Chakrabarti, S. Suwas, és M. Arora, Szerk., in Lecture Notes in Mechanical Engineering. , Singapore: Springer Nature Singapore, 2023, o. 73–80. doi: 10.1007/978-981-19-0561-2_7.
- [19] Abhijit Gosavi és S. E. Grasman, „Simulation-based optimization for determining AGV capacity in a manufacturing system”, 2009, doi: 10.13140/2.1.1059.3928.
- [20] A. Negahban és J. S. Smith, „Simulation for manufacturing system design and operation: Literature review and analysis”, Journal of Manufacturing Systems, köt. 33, sz. 2, o. 241–261, ápr. 2014, doi: 10.1016/j.jmsy.2013.12.007.
- [21] J. Fu, J. Zhang, G. Ding, S. Qin, és H. Jiang, „Determination of vehicle requirements of AGV system based on discrete event simulation and response surface methodology”, Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, köt. 235, sz. 9, o. 1425–1436, júl. 2021, doi: 10.1177/09544405421995602.
- [22] A. Jahed és R. Tavakkoli-Moghaddam, „Mathematical modeling for a flexible manufacturing scheduling problem in an intelligent transportation system”, Iran J Manag Stud, sz. Online First, 0 2020, doi: 10.22059/ijms.2020.261618.673203.
- [23] R. Ojstersek, B. Acko, és B. Buchmeister, „Simulation Study of a Flexible Manufacturing System Regarding Sustainability”, Int. j. simul. model., köt. 19, sz. 1, o. 65–76, márc. 2020, doi: 10.2507/IJSIMM19-1-502.
- [24] T. B. Ionescu, „Leveraging Graphical User Interface Automation for Generic Robot Programming”, Robotics, köt. 10, sz. 1, o. 3, dec. 2020, doi: 10.3390/robotics10010003.

- [25] M. Jahangirian, T. Eldabi, A. Naseer, L. K. Stergioulas, és T. Young, „Simulation in manufacturing and business: A review”, *European Journal of Operational Research*, köt. 203, sz. 1, o. 1–13, máj. 2010, doi: 10.1016/j.ejor.2009.06.004.
- [26] N. Brahimi, A. Dolgui, E. Gurevsky, és A. R. Yelles-Chaouche, „A literature review of optimization problems for reconfigurable manufacturing systems”, *IFAC-PapersOnLine*, köt. 52, sz. 13, o. 433–438, 2019, doi: 10.1016/j.ifacol.2019.11.097.
- [27] K. Singh és S. Sundar, „A hybrid genetic algorithm for the degree-constrained minimum spanning tree problem”, *Soft Comput*, köt. 24, sz. 3, o. 2169–2186, febr. 2020, doi: 10.1007/s00500-019-04051-x.
- [28] A. R. Yelles-Chaouche, E. Gurevsky, N. Brahimi, és A. Dolgui, „Reconfigurable manufacturing systems from an optimisation perspective: a focused review of literature”, *International Journal of Production Research*, köt. 59, sz. 21, o. 6400–6418, nov. 2021, doi: 10.1080/00207543.2020.1813913.
- [29] T. Navalertporn és N. V. Afzulpurkar, „Optimization of tile manufacturing process using particle swarm optimization”, *Swarm and Evolutionary Computation*, köt. 1, sz. 2, o. 97–109, jún. 2011, doi: 10.1016/j.swevo.2011.05.003.
- [30] Q.-K. Pan, M. Fatih Tasgetiren, P. N. Suganthan, és T. J. Chua, „A discrete artificial bee colony algorithm for the lot-streaming flow shop scheduling problem”, *Information Sciences*, köt. 181, sz. 12, o. 2455–2468, jún. 2011, doi: 10.1016/j.ins.2009.12.025.
- [31] J. M. Jauregui Becker, J. Borst, és A. Van Der Veen, „Improving the overall equipment effectiveness in high-mix-low-volume manufacturing environments”, *CIRP Annals*, köt. 64, sz. 1, o. 419–422, 2015, doi: 10.1016/j.cirp.2015.04.126.
- [32] Dunarea de Jos University of Galati, Industrial Engineering Department 800201, Romania, C. Afteni, és G. Frumuşanu, „A Review on Optimization of Manufacturing Process Performance”, *IJMO*, köt. 7, sz. 3, o. 139–144, jún. 2017, doi: 10.7763/IJMO.2017.V7.573.
- [33] T. A. M. Tolio, L. Monostori, J. Váncza, és O. Sauer, „Platform-based manufacturing”, *CIRP Annals*, köt. 72, sz. 2, o. 697–723, 2023, doi: 10.1016/j.cirp.2023.04.091.
- [34] H. ElMaraghy, L. Monostori, G. Schuh, és W. ElMaraghy, „Evolution and future of manufacturing systems”, *CIRP Annals*, köt. 70, sz. 2, o. 635–658, 2021, doi: 10.1016/j.cirp.2021.05.008.
- [35] Z. Vámosy, *Automatizált Eszközök*. Budapest: Budapesti Műszaki Főiskola, 2002.
- [36] M. Shneier és R. Bostelman, „Literature Review of Mobile Robots for Manufacturing”, *National Institute of Standards and Technology, NIST IR 8022*, máj. 2015. doi: 10.6028/NIST.IR.8022.

- [37] Á. Cservenák és J. Husár, „A Multidisciplinary Learning Model Using AGV and AMR for Industry 4.0/5.0 Laboratory Courses: A Study”, *Applied Sciences*, köt. 14, sz. 17, o. 7965, szept. 2024, doi: 10.3390/app14177965.
- [38] R. Jánoš és mtsai., „Stability and Dynamic Walk Control of Humanoid Robot for Robot Soccer Player”, *Machines*, köt. 10, sz. 6, o. 463, jún. 2022, doi: 10.3390/machines10060463.
- [39] J. Husár és L. Knapčíková, „ONLINE AND OFFLINE CONTROL OF COLLABORATIVE ROBOTS USED MIXED REALITY”, *AT*, köt. 7, sz. 2, o. 61–66, jún. 2021, doi: 10.22306/atec.v7i2.109.
- [40] S. Saniuk, S. Grabowska, és M. Straka, „Identification of Social and Economic Expectations: Contextual Reasons for the Transformation Process of Industry 4.0 into the Industry 5.0 Concept”, *Sustainability*, köt. 14, sz. 3, o. 1391, jan. 2022, doi: 10.3390/su14031391.
- [41] J. Trojanowska, K. Zywicki, M. L. R. Varela, és J. M. Machado, „Shortening changeover time — An industrial study”, in 2015 10th Iberian Conference on Information Systems and Technologies (CISTI), Aveiro, Portugal: IEEE, jún. 2015, o. 1–6. doi: 10.1109/CISTI.2015.7170373.
- [42] H. Martínez-Barberá és D. Herrero-Pérez, „Autonomous navigation of an automated guided vehicle in industrial environments”, *Robotics and Computer-Integrated Manufacturing*, köt. 26, sz. 4, o. 296–311, aug. 2010, doi: 10.1016/j.rcim.2009.10.003.
- [43] J. Kramer és M. Scheutz, „Development environments for autonomous mobile robots: A survey”, *Auton Robot*, köt. 22, sz. 2, o. 101–132, jan. 2007, doi: 10.1007/s10514-006-9013-8.
- [44] J. Huang, S. Junginger, H. Liu, és K. Thurow, „Indoor Positioning Systems of Mobile Robots: A Review”, *Robotics*, köt. 12, sz. 2, o. 47, márc. 2023, doi: 10.3390/robotics12020047.
- [45] A. Li, J. Cao, S. Li, Z. Huang, J. Wang, és G. Liu, „Map Construction and Path Planning Method for a Mobile Robot Based on Multi-Sensor Information Fusion”, *Applied Sciences*, köt. 12, sz. 6, o. 2913, márc. 2022, doi: 10.3390/app12062913.
- [46] J. Yang, X. Wang, és P. Bauer, „V-Shaped Formation Control for Robotic Swarms Constrained by Field of View”, *Applied Sciences*, köt. 8, sz. 11, o. 2120, nov. 2018, doi: 10.3390/app8112120.
- [47] L. Cheng, X. Wu, és Y. Wang, „Artificial Flora (AF) Optimization Algorithm”, *Applied Sciences*, köt. 8, sz. 3, o. 329, febr. 2018, doi: 10.3390/app8030329.
- [48] H. Wang, Y. Li, T. Chang, S. Chang, és Y. Fan, „Event-Driven Sensor Deployment in an Underwater Environment Using a Distributed Hybrid Fish Swarm Optimization Algorithm”, *Applied Sciences*, köt. 8, sz. 9, o. 1638, szept. 2018, doi: 10.3390/app8091638.
- [49] P. Kormushev, S. Calinon, és D. Caldwell, „Reinforcement Learning in Robotics: Applications and Real-World Challenges”, *Robotics*, köt. 2, sz. 3, o. 122–148, júl. 2013, doi: 10.3390/robotics2030122.

- [50] C. Follini és mtsai., „BIM-Integrated Collaborative Robotics for Application in Building Construction and Maintenance”, *Robotics*, köt. 10, sz. 1, o. 2, dec. 2020, doi: 10.3390/robotics10010002.
- [51] H.-G. Kim és mtsai., „Development of a Multi-Robot System for Pier Construction”, *Machines*, köt. 12, sz. 6, o. 385, jún. 2024, doi: 10.3390/machines12060385.
- [52] V. B. Justo, A. Gupta, T. F. Umland, és D. Göhlich, „Minimum Energy Utilization Strategy for Fleet of Autonomous Robots in Urban Waste Management”, *Robotics*, köt. 12, sz. 6, o. 159, nov. 2023, doi: 10.3390/robotics12060159.
- [53] I. Lee, „Service Robots: A Systematic Literature Review”, *Electronics*, köt. 10, sz. 21, o. 2658, okt. 2021, doi: 10.3390/electronics10212658.
- [54] T. Breuer és mtsai., „Johnny: An Autonomous Service Robot for Domestic Environments”, *J Intell Robot Syst*, köt. 66, sz. 1–2, o. 245–272, ápr. 2012, doi: 10.1007/s10846-011-9608-y.
- [55] L. Emmi, M. Gonzalez-de-Soto, G. Pajares, és P. Gonzalez-de-Santos, „New Trends in Robotics for Agriculture: Integration and Assessment of a Real Fleet of Robots”, *The Scientific World Journal*, köt. 2014, o. 1–21, 2014, doi: 10.1155/2014/404059.
- [56] J. Alonso-Mora, A. Breitenmoser, M. Rufli, R. Siegwart, és P. Beardsley, „Image and animation display with multiple mobile robots”, *The International Journal of Robotics Research*, köt. 31, sz. 6, o. 753–773, máj. 2012, doi: 10.1177/0278364912442095.
- [57] M. Boleraczki és D. Dr. Fodor, „Robotmegfogó szerkezet optimális kialakítása additív gyártáshoz”, *SzámOkt 2020 – 30th International Conference on Computers and Education*, o. 53–56, okt. 2020.
- [58] M. Boleraczki, I. G. Gyurika, és D. Fodor, „Gripper Finger Design for Special Purpose Applications”, *Hung. J. Ind. Chem.*, köt. 49, sz. 2, o. 91–95, 2021, doi: 10.33927/hjic-2021-28.
- [59] G.-Z. Yang és mtsai., „The grand challenges of Science Robotics”, *Sci. Robot.*, köt. 3, sz. 14, o. eaar7650, jan. 2018, doi: 10.1126/scirobotics.aar7650.
- [60] H. Tan, „A brief history and technical review of the expert system research”, *IOP Conf. Ser.: Mater. Sci. Eng.*, köt. 242, o. 012111, szept. 2017, doi: 10.1088/1757-899X/242/1/012111.
- [61] M. Ashraf és F. Hasan, „Configuration selection for a reconfigurable manufacturing flow line involving part production with operation constraints”, *Int J Adv Manuf Technol*, köt. 98, sz. 5–8, o. 2137–2156, szept. 2018, doi: 10.1007/s00170-018-2361-7.
- [62] M. M. Mabkhot, S. Darmoul, A. M. Al-Samhan, és A. Badwelan, „A Multi-Criteria Decision Framework Considering Different Levels of Decision-Maker Involvement to Reconfigure Manufacturing Systems”, *Machines*, köt. 8, sz. 1, o. 8, febr. 2020, doi: 10.3390/machines8010008.
- [63] A. L. Szejka, O. Canciglieri Junior, és F. Mas, „Knowledge-based expert system to drive an informationally interoperable manufacturing system: An

- experimental application in the Aerospace Industry”, *Journal of Industrial Information Integration*, köt. 41, o. 100661, szept. 2024, doi: 10.1016/j.jii.2024.100661.
- [64] M. Ritou, F. Belkadi, Z. Yahouni, C. Da Cunha, F. Laroche, és B. Furet, „Knowledge-based multi-level aggregation for decision aid in the machining industry”, *CIRP Annals*, köt. 68, sz. 1, o. 475–478, 2019, doi: 10.1016/j.cirp.2019.03.009.
- [65] A. Al-Mahturi, F. Santoso, M. A. Garratt, és S. G. Anavatti, „A Novel Evolving Type-2 Fuzzy System for Controlling a Mobile Robot under Large Uncertainties”, *Robotics*, köt. 12, sz. 2, o. 40, márc. 2023, doi: 10.3390/robotics12020040.
- [66] Q. Cao és mtsai., „KSPMI: A Knowledge-based System for Predictive Maintenance in Industry 4.0”, *Robotics and Computer-Integrated Manufacturing*, köt. 74, o. 102281, ápr. 2022, doi: 10.1016/j.rcim.2021.102281.
- [67] M. Vuković és S. Thalmann, „Causal Discovery in Manufacturing: A Structured Literature Review”, *JMMP*, köt. 6, sz. 1, o. 10, jan. 2022, doi: 10.3390/jmmp6010010.
- [68] S. I. Shafiq, C. Sanin, és E. Szczerbicki, „Knowledge-Based Virtual Modeling and Simulation of Manufacturing Processes for Industry 4.0”, *Cybernetics and Systems*, köt. 51, sz. 2, o. 84–102, febr. 2020, doi: 10.1080/01969722.2019.1705546.
- [69] M. Janiček, R. Ružarovský, K. Velíšek, és R. Holubek, „Analysis of voice control of a collaborative robot”, *J. Phys.: Conf. Ser.*, köt. 1781, sz. 1, o. 012025, febr. 2021, doi: 10.1088/1742-6596/1781/1/012025.
- [70] D. R. Delgado Sobrino, V. Rychtarik, és D. Cagáňová, „Digital Twin design at the material flow level”, *J. Phys.: Conf. Ser.*, köt. 2212, sz. 1, o. 012012, febr. 2022, doi: 10.1088/1742-6596/2212/1/012012.
- [71] M. Boleraczki, C. Sik-Lanyi, Z. Zsák, és T. Papp, „DEVELOPING THE SUPPORT FRAMEWORK SYSTEM OF SPECIAL PURPOSE MACHINES DESIGNING”, *IOP Conf. Ser.: Mater. Sci. Eng.*, köt. 448, o. 012029, nov. 2018, doi: 10.1088/1757-899X/448/1/012029.
- [72] L. A. Simonova és G. M. Kapitonova, „Frame model of knowledge representation in the expert system of technological route selection”, *IOP Conf. Ser.: Mater. Sci. Eng.*, köt. 915, sz. 1, o. 012054, szept. 2020, doi: 10.1088/1757-899X/915/1/012054.
- [73] B. Burcu és B. Gülcin, „Robot Selection for Warehouses”, XIV. International Logistics and Supply Chain Congress, 0 2016.
- [74] Y. Fu, M. Li, H. Luo, és G. Q. Huang, „Industrial robot selection using stochastic multicriteria acceptability analysis for group decision making”, *Robotics and Autonomous Systems*, köt. 122, o. 103304, dec. 2019, doi: 10.1016/j.robot.2019.103304.
- [75] International Trade and Logistics Department, Sivas Cumhuriyet University, 58140 (Turkey) és mtsai., „Selection of insulation materials with

- PSI-CRITIC based CoCoSo method”, *rdlc*, köt. 20, sz. 2, o. 382–392, aug. 2021, doi: 10.7764/RDLC.20.2.382.
- [76] D. Diakoulaki, G. Mavrotas, és L. Papayannakis, „Determining objective weights in multiple criteria problems: The critic method”, *Computers & Operations Research*, köt. 22, sz. 7, o. 763–770, aug. 1995, doi: 10.1016/0305-0548(94)00059-H.
- [77] A. Tarafdar, A. Shaikh, M. N. Ali, és A. Haldar, „An integrated fuzzy decision-making framework for autonomous mobile robot selection: balancing subjective and objective measures with fuzzy TOPSIS and picture fuzzy CoCoSo approach”, *Journal of the Operational Research Society*, o. 1–27, ápr. 2025, doi: 10.1080/01605682.2025.2486705.
- [78] C. Kao és S.-T. Liu, „Group decision making in data envelopment analysis: A robot selection application”, *European Journal of Operational Research*, köt. 297, sz. 2, o. 592–599, márc. 2022, doi: 10.1016/j.ejor.2021.05.013.
- [79] S. Opricovic és G.-H. Tzeng, „Compromise solution by MCDM methods: A comparative analysis of VIKOR and TOPSIS”, *European Journal of Operational Research*, köt. 156, sz. 2, o. 445–455, júl. 2004, doi: 10.1016/S0377-2217(03)00020-1.
- [80] V. Chodha, R. Dubey, R. Kumar, S. Singh, és S. Kaur, „Selection of industrial arc welding robot with TOPSIS and Entropy MCDM techniques”, *Materials Today: Proceedings*, köt. 50, o. 709–715, 2022, doi: 10.1016/j.matpr.2021.04.487.
- [81] S. S. Goswami és mtsai., „Analysis of a Robot Selection Problem Using Two Newly Developed Hybrid MCDM Models of TOPSIS-ARAS and COPRAS-ARAS”, *Symmetry*, köt. 13, sz. 8, o. 1331, júl. 2021, doi: 10.3390/sym13081331.
- [82] M. Boleraczki és I. G. Gyurika, „Mobile robot models for manufacturing systems”, előadás 6th World Congress on Mechanical, Chemical, and Material Engineering, aug. 2020. doi: 10.11159/icmie20.135.
- [83] M. Boleraczki, „Development of a knowledge base for an expert system supporting the decision of industrial mobile robot selection”, *J. Phys.: Conf. Ser.*, köt. 2714, sz. 1, o. 012004, febr. 2024, doi: 10.1088/1742-6596/2714/1/012004.

Mellékletek

Programkód a mobil robot központú gyártórendszer első esettanulmányához,
70. oldal

```
while state != 8:
    if state == 1:
        state = 3
        robot_all += 1
    else:
        state = 8
    if state == 2:
        B_packages += 1
        robot_all += 1
        if charge > 20: //check the battery
            state = 4
        else:
            state = 5
    if state == 3:
        if robot_i == path_length_AB:
            state = 2
            robot_i = 0
            robot_move += 1
        else:
            robot_i += 1
            charge -= 1
            robot_move += 1
    if state == 4:
        if robot_i == path_length_BA:
            state = 1
            robot_i = 0
            robot_move += 1
        else:
            robot_i += 1
            charge -= 1
            robot_move += 1
    if state == 5:
        if robot_i == path_length_charging:
            state = 6
            robot_i = 0
        else:
            robot_i += 1
            charge -= 1
    if state == 6:
        if charge >= max_charge-5:
            state = 7
            robot_i = 0
            robot_all += 1
            robot_charge_no += 1
        else:
            robot_i += 1
            charge += 5
            robot_all += 1
    if state == 7:
        if robot_i == path_length_charging:
            state = 1
            robot_i = 0
        else:
            robot_i += 1
        if A_packages > 0: // if there is available package
            A_packages -= 1 // count down the number of the package
```

```

charge -= 1
increment i // next time step

```

Állapotok a fenti kódhoz:

```

1: pick up the package
2: put down the package
3: robot moving from A to B
4: robot moving from B to A
5: robot moving from B to charging point
6: robot is charging
7: robot is moving from charging point to A
8: simulation over
i = 0 // cycle variable, using for time step
path_length_AB = 10 // the length of the path from A to B
path_length_BA = 10 // the length of the path from B to A
path_length_charging = 6 // the length of the path from B to the charging point, and from charging point to A
actual_state = 1 // starting state
packages = 1000 // how many packages are ready to move
charge = 100 // this is the capacity of the battery
robot_charge_no = 0 // charging cycle number

```

CLIPS tények (részlet)

```

(deftemplate robot
(slot id)
(slot manufacturer)
(slot type)
(slot loading_capacity)
(slot inside)
(slot outside)
(slot robot_weight)
(slot battery_capacity)
(slot time_battery_load)
(slot towing)
(slot ground_load)
(slot max_velocity)
(slot pallett_move)
(slot esd_area)
(slot cleanroom)
(slot machine_number)
(slot movements)
(slot robot_numbers)
(slot reliability)
(slot performance)
)
(deffacts robot-data
(robot (id R0)
(manufacturer KUKA)
(type 5)
(loading_capacity 150.0)
(inside True)
(outside False)
(robot_weight 244)
(battery_capacity 30.0)
(time_battery_load 120.0)
(towing False)
(ground_load False)
(max_velocity 3)
(pallett_move False)
(esd_area False)
(cleanroom False)
(machine_number high)
(movements long)
(robot_numbers extra_middle)
(reliability 0)
(performance 75))

```

Köszönetnyilvánítás

Elsősorban hálásan köszönöm az Úrnak a lelki erőt ennek a kutatásnak a befejezéséhez és a dolgozat megírásához.

Köszönöm témavezetőmnek, Dr. Gyurika Istvánnak a sok támogatást, és a fókuszban tartást, valamint a határidők betartására való figyelmeztetést, iránymutatást és a publikálásokban nyújtott segítséget.

Köszönöm feleségemnek, Mariannak, lányaimnak, Mírának és Julcsinak a nyugodt családi háttér által nyújtott pótolhatatlan hozzájárulásukat.

Köszönöm a Pannon Egyetem Géptan Intézeti Tanszék minden munkatársának, kedves kollégáimnak a folyamatos biztatást és támogatást.

Köszönöm édesapámnak és édesanyámnak a lehetőséget a tanulmányaim végzésére.

Köszönöm lelki vezetőmnek, János atyának, hogy a legnehezebb időkben is mellém állt, és útmutatásával továbblendített akkor is, amikor már feladtam volna.

Köszönöm bírálóimnak a lelkiismeretesen végzett bírálói tevékenységet, és hasznos tanácsaikat, amivel hozzájárultak a dolgozat értékének növeléséhez.

„Íme egy kicsiny tűz milyen nagy erdő felgyújt!”

Jakab 3:5