

MEZŐGAZDASÁGI TERMÉKEK FELVÁSÁRLÁSI FOLYAMATÁNAK SZIMULÁCIÓJA, KÜLÖNÖS TEKINTETTEL A CUKORRÉPÁRA

OTKA Kutatási téma 2002 – 2005. Nyilvántartási szám: T0 37555

TARTALOMJEGYZÉK

| | |
|--|----|
| 1. Kutatási célkitűzések | 2 |
| 2. Történeti visszapillantás a szakirodalom tükrében..... | 2 |
| 2.1. A cukorrépa és a cukor története | 3 |
| 2.2. A cukorgyártás története..... | 3 |
| 2.3. A magyar cukoripar és cukorrépa termesztés története..... | 3 |
| 2.4. A cukorrépa betakarítás gépesítésének fejlődése | 3 |
| 2.5. A cukorrépa gazdasági értéke, a répatermesztés jövője | 4 |
| 3. A cukorrépa termesztés és cukorgyártás helyzete és várható változásai..... | 4 |
| 3.1. A cukorrépa termesztés és cukorgyártás hazai helyzete..... | 4 |
| 3.2. Az Európai Unió cukorágazatának helyzete és jellemzői | 4 |
| 3.3. A cukorrépa termesztés és cukorgyártás területén várható változások | 5 |
| 4. A cukorrépa betakarítással, tárolással, szállítással kapcsolatos megfigyelések és javaslatok..... | 5 |
| 4.1. A cukorrépa termeltetés, beszállítás és átvétel hazai és európai gyakorlata | 6 |
| 4.1.1. A cukorrépa felvásárlási és átvételi folyamat a Szerencsi Cukorgyár Rt-nél..... | 6 |
| 4.1.2. Az integrált répalogisztikai rendszer felépítése és működése | 7 |
| 4.1.3. A cukorrépa beszállítás módszereinek rendszerezése | 8 |
| 5. Szántóföldi mérések | 9 |
| 5.1. A mérések leírása..... | 9 |
| 5.1.1. A répa kiszedése és prizmázása..... | 10 |
| 5.1.2. A répa felszedése, tisztítása, rakodása és szállítása..... | 10 |
| 5.1.3. A répa vagonba rakása..... | 11 |
| 5.1.4. A fogások hosszának számítása..... | 11 |
| 5.1.5. A pótkocsis szerelvények számának meghatározása..... | 11 |
| 5.2. A megfigyelt és mért adatok..... | 13 |
| 5.2.1. A répa kiszedés és prizmázás mérése | 13 |
| A betakarítógép megfigyelése | 14 |
| A szállítójárművek megfigyelése a prizmánál (betakarítás)..... | 14 |
| 5.2.2. A répa felszedés, tisztítás, rakodás és szállítás mérése..... | 14 |
| A tisztító-rakodó gép megfigyelése | 15 |
| A homlok-rakodó gép megfigyelése a prizmánál..... | 15 |
| A szállítójárművek megfigyelése a prizmánál (szállítás)..... | 15 |
| A szállítójárművek megfigyelése a vasúti átadóhelyen..... | 15 |
| 5.2.3. A répa vagonba rakás mérése | 16 |
| 5.3. A mérési eredmények feldolgozása..... | 16 |
| 6. A cukorrépa betakarítás és szállítás vizsgálata szimulációval | 17 |
| 6.1. Bevezetés | 17 |
| 6.2. Szimulációs modellek és nyelvek..... | 17 |
| 6.3. Az Arena modellezési környezet..... | 18 |
| 6.4. Modellépítés az Arena környezetben | 20 |
| 6.4.1. Elemek (elements) és tulajdonságok (properties)..... | 20 |
| 6.4.2. Folyamatábra és adatmodulok | 21 |

| | |
|---|----|
| 6.5. A cukorrépa betakarítás logikai modellje és paraméterei..... | 21 |
| 6.5.1. A betakarítás almodellje | 22 |
| 6.5.2. A szállítás almodellje | 25 |
| 6.5.3. A rakodás almodellje | 26 |
| 6.6. A cukorrépa betakarítási modell futtatása | 28 |
| 6.6.1. A szimuláció paramétereinek beállítása | 28 |
| 6.6.2. A modell ellenőrzése | 29 |
| A cukorrépa betakarítás és szállítás szimulációs modelljének input adatai | 29 |
| Erőforrások | 29 |
| 6.6.3. Az eredmények megtekintése és értelmezése..... | 29 |
| 6.6.4. Az eredmények értékelése | 31 |
| 7. Cukorrépa átvevő depók számának és helyének meghatározása | 34 |
| 7.1. Bevezetés | 34 |
| 7.2. A fokozott tisztítású rakodás és szállítás | 34 |
| 7.3. Modell a depók optimális helyének kijelölésére | 35 |
| 7.4. Modell alkalmazása | 38 |
| 7.4.1. A GPS használata és az adatok gyűjtése | 38 |
| 8. Összefoglalás..... | 40 |
| Irodalom | 42 |

A zárójelentést összeállították:

| | |
|--------------|--------------------------------|
| 1-3. fejezet | dr. Soós Pál, dr. Balogh Antal |
| 4-8. fejezet | dr. Benkő János |

A modelleket és a számítógép programokat dr. Benkő János készítette.

A zárójelentés teljes terjedelemben a

<http://www.gek.szie.hu/tanszekek/geti/logisztika/lapok/Cjelentes.htm> címen olvasható.

1. KUTATÁSI CÉLKITŰZÉSEK

A cukorrépa betakarítása összehangolt szervezést igénylő energia- és költségigényes művelet. A cél az, hogy a megtermelt répa a lehető legkisebb veszteség mellett és legkisebb költséggel kerüljön a cukorgyárba. A mezőgazdasági termelés sajátosságaiból fakadóan azonban (időszaktól, időjárástól való függőség) a betakarítás megfelelő ütemezése, a nagy mennyiségű alapanyag megfelelő időben és minőségben való beszállításának megtervezése, illetve a folyamatok optimális összehangolása bonyolult feladat. Az ilyen összetett, nagyszámú befolyásoló tényezővel és véletlen hatásokkal átfont rendszer vizsgálatának hatékony eszköze a szimuláció. A kutatási munka célkitűzése a cukorrépa felvásárlási folyamatának szimulációjával feltárni a betakarítás, a tárolás és a szállítás folyamatában a szűk keresztmetszeteket, melynek ismeretében eldönthető, hogy hol, milyen változtatásokra, műszaki és technológiai fejlesztésekre van szükség ahhoz, hogy idő- és energia- ráfordítás tekintetében hatékonyabb legyen a „répalogisztika”.

2. TÖRTÉNETI VISSZAPILLANTÁS A SZAKIRODALOM TÜKRÉBEN

A cukorrépa termeltetési és felvásárlási folyamatának vizsgálatához elengedhetetlen az adott tématerület mélyrehatóbb és szélesebb körű megismerése, ezért kutatómunkánk első részében hazai és külföldi szakirodalmi munkák alapján áttekintést adunk a cukorrépa, a cukorrépa termesztés és a cukorgyártás történetéről.

2.1. A cukorrépa és a cukor története

A cukorrépa az egyik legfiatalabb kultúrnövényünk, nemesítését, termesztését és a répacukor gyártását csak a 19. század elején kezdték meg. A cukorrépát azonban már jóval korábban, ie. kb. 2000 évvel ismerték és egyes változatait étkezési, gyógyászati célra és állati eledelül termesztették. A mai cukorrépának a közvetlen őse az Achard által kiválasztott sziléziai fehérrépa. A répában lévő fehércukrot Marggraf fedezte fel 1747-ben, aki megállapította, hogy az a nádból kinyerhető cukorral azonos minőségű. A 19. század elejétől kezdve Európában a cukor- és takarmányrépát már általánosan termesztik.

A cukor szülőhazája India, ahol a szilárdabb cukrot már az i. u. 300 körüli időben ismerték, s innen terjedt el Kínába, Perzsiába és Egyiptomba. A nádméz ismeretét a keresztes hadak terjesztették el, s Velence hozta forgalomba. A nádmézből cukorkristályok váltak ki, ha jobban beszáradt, később ezeket a kristályokat tudatosan állították elő, így keletkezett a cukorgyártás. Magyarországon 1419-ből találjuk az első feljegyzést a cukorról, Zsigmond király udvarában. A 16. században a cukor már szélesebb körben elterjedt, de drágasága miatt sokak számára még elérhetetlen volt. Csak az 1800-as évek közepétől, a répacukor gyártás megindulása után vált a cukor általánosan fogyasztott és elérhető áru élelmiszerré.

2.2. A cukorgyártás története

Marggráf felfedezésének jelentőségét tanítványa, Achard ismerte fel, s ő hozta létre 1802-ben Sziléziában az első répacukor gyárat. A répából való cukorgyártás azonban ekkor még nem volt versenyképes - a répa kevés cukrot tartalmazott, és a cukornak répaíze volt – így Achard gyára csakhamar megszűnt. A cukorgyártás csak 1825-től indult fejlődésnek, mikor a mésszel való elválasztás ismeretessé lett, mert ettől kezdve az eljárás gazdaságosabbá vált. A cukorgyártás Franciaországból indult el s ott is tökéletesedett az idők folyamán, s az utolsó ötven évben a cukorgyártás óriási léptekkel haladt előre.

2.3. A magyar cukoripar és cukorrépa termesztés története

Magyarországon Lacsny létesíti az első cukorgyárat Nagyfödémesen (1830), amelyet további 63 kisebb - nagyobb gyár építése követett. Az 1888. utáni években további korszerű gyárak épültek és a cukorgyártás lassan a teljes hazai szükségletet fedezte, sőt exportra is került. A hazai gyárakban ekkor a világ valamennyi répacukor gyára közül a legnagyobb átlagtermelést érték el. A répa termesztésében a cukorgyári gazdaságok jártak az élen, az itt szerzett tapasztalatok elterjedtek a répatermesztő gazdák körében.

Az első világháború és az azt követő válságok komoly megrázkódtatást okoztak a cukoriparnak is: a háború előtti Magyarországon 30, a trianoni Magyarországon már mindössze csak 12 cukorgyár működött. A második világháborúban a magyar cukoripart is sok kár érte, de az 1948 körüli időszakra már helyreállt a termelés. A cukorgyárak államosítása, a tervszerű központi vezetés tette lehetővé az egyes gyárak további korszerűsítését.

2.4. A cukorrépa betakarítás gépesítésének fejlődése

A cukorrépát kézi erővel, részben- vagy teljesen gépesítve szedik. A cukorrépa hagyományos termesztési módjánál a betakarítás minden fázisát (fejezést, kiszedést, kocsira rakás) kézzel végezték, majd fogatos, később traktoros vontatású fejező-lazító szerszámokkal könnyítették meg. A részben gépesített cukorrépa betakarítás gépei két fő csoportba sorolhatók: az egyik típus a répát a földben fejeli, és a gyökerét ezt követően szedi, a másik típus répát kilazítja a földből, levélzeténél fogva kiemeli és a gépen fejezi.

Ez a részleges gépesítés azonban még mindig sok kézi erőt kötött le, és kis teljesítménye miatt nem elégítette ki a növekvő igényeket. A problémát a betakarítás teljes gépesítése

oldotta meg, amit a traktorok teljesítményének növekedése tett lehetővé. A teljesen gépesített cukorrépa betakarításhoz hazánkban a hatvanas évek elején kétféle géptípus állt rendelkezésre: cukorrépa betakarítás földben fejező-gépekkel, illetve cukorrépa betakarítás a gépen végzett fejezéssel. A gépek többsége az előző csoporthoz tartozott.

2.5. A cukorrépa gazdasági értéke, a répatermesztés jövője

A cukorrépa gazdasági értékét és jelentőségét az adja egyrészt, hogy szinte a teljes termése hasznosítható, a cukor kivonása után maradó rész kitűnő takarmányozási célra, másrészt a mezőgazdasági termények között a cukorrépa területegységre vonatkoztatott energiataralma a legnagyobb, de jelentős szerepe volt az intenzív mezőgazdasági kultúra fellendítésében is.

3. A CUKORRÉPA TERMESZTÉS ÉS CUKORGYÁRTÁS HELYZETE ÉS VÁRHATÓ VÁLTOZÁSAI

3.1. A cukorrépa termesztés és cukorgyártás hazai helyzete

A rendszerváltás követően a megváltozott gazdasági körülmények, a szerkezetváltás, a termelői és ipari privatizáció hatásaként a hazai cukorvertikum is átalakult. Ez az átrendeződés kezdetben nem ment zökkenőmentesen, a szaktudás és tőke nélküli kényszer-vállalkozók megjelenésével a termelők száma a korábbihoz képest megsokszorozódott, ami a termelés színvonalának csökkenéséhez, a minőségi paraméterek romlásához, az ágazat jövedelmezőségének csökkenéséhez vezetett. Az 1990-es évek közepétől megindult szelekció nyomán kialakult egy jóval kisebb, de nagyobb anyagi- és szellemi felkészültséggel rendelkező, stabil cukorrépa-termesztői kör, amely képessé vált az EU elvárásoknak megfelelően termelni (számuk ma az integrátorokkal együtt 752, kb. 60 ezer hektár területtel).

A jelenlegi struktúra gyakorlatilag 1997-re alakult ki, napjainkban három társaság 6 cukorgyárat működtet, s a tulajdonosok az EU legnagyobb cukorgyártói. A gyárak többségének kapacitása 6500 - 10000 t/nap közötti, az átlagos kapacitás 6700 t/nap, ami jelentős növekedést mutat az 1989. évi 3825 t/nap kapacitáshoz képest.

A cukorrépa termesztés és cukorgyártás hazai helyzetének jobb megismerése érdekében szakirodalmi adatok és üzemi tapasztalatok alapján részletesen - a vetéstől a betakarításig - átfogóan elemeztük a 2002. évi hazai cukorrépa termesztés főbb adatait.

3.2. Az Európai Unió cukorágazatának helyzete és jellemzői

Az EU cukortermelésének 98%-a répacukor, s a bővítés során belépett új tagok (köztünk hazánk) tovább növelték a répacukor európai fölényét. A 15 tagállamban 1,8 - 2 millió hektáron termesztenek cukorrépát, amely a teljes szántóföldi vetésterület kevesebb, mint 2% -a. A megtermelt 100 - 115 millió tonna cukorrépából évente 15 - 18 millió tonna fehér cukrot állítanak elő, emellett az EU fogyasztása 12 - 13 millió tonna.

Az EU az 1968 óta működő cukorpiaci rendtartásnak köszönhetően védte és védi továbbra is cukorpiacát a világpiacon ellen, ami kisebb módosításokkal ma is a kezdeti elvek szerint működik. Az EU cukorrendtartásának lényege az önellátás, ennek érdekében a termelést kvóta rendszerrel szabályozzák. Az EU cukorpiaci rendtartása napjainkra a kritikák kereszttüzébe került, a világpiacon változások és átrendeződések, valamint a Közösségben rejlő belső feszültségek miatt elengedhetetlenné vált az EU cukorágazatának radikális megreformálása.

3.3. A cukorrépa termesztés és cukorgyártás területén várható változások

A cukor hozzávetőlegesen 130 millió tonnás világpiacát alapvetően befolyásolja egyrészt, hogy az önellátásra képes országok saját cukorvertikum fenntartására törekednek nagyobb fokú piaci védelmet alkalmazva, másrészt a trópusi, szubtrópusi országokban termelt cukornádból lényegesen olcsóbban lehet a cukrot előállítani, mint cukorrépából. Ez a kettősség szüli az export dömpinget, a kereskedelem-politikai vitákat.

A nádcukor aránya a teljes világtermelésből 1995-ben átlépte a 70 %-t, a valós térnyerés viszont jóval számottevőbb, mivel a 80-as években 80-100 millió tonnás cukortermelés mára 130 millió tonnára nőtt. A világpiac karakterisztikáit tehát az exportra termelt nádcukor határozza meg, ami lenyomja az árakat. Jelenleg egy 1999-ben elkezdődött ciklus van folyamatban, amely azonban az eddigiekhez képest a jelentős brazil termelésbővítés miatt tovább tarthat.

Az EU-, illetve Magyarország cukoripara számára az alapvető kihívást ezen a világpiacon változások, átrendeződések jelentik, melyek miatt elengedhetlenné vált az EU cukorágazatának radikális megreformálása, a cukorrendtartás átalakítása. Az Európai Unió agrárminiszterei 2005. novemberében megállapodást kötöttek a cukorpiac közös szervezésének széleskörű reformjáról. A reform legfőbb célja az ágazat fejlődésének és hosszú távú fennmaradásának elősegítése, versenyképességének és piacorientáltságának javítása, valamint a piaci egyensúly biztosítása

A reform 2014/2015-ig felülvizsgálati záradék nélkül határozza meg az európai cukorágazat gazdasági és jogi keretét. A 36%-os árcsökkenést egy 4 évig működő nagyvonalú szerkezetátalakítási alap kíséri. A szerkezetátalakítási alap fő céljai: annak ösztönzése, hogy a kevésbé versenyképes termelők hagyják el az ágazatot, továbbá pénz biztosítása annak érdekében, hogy meg lehessen birkózni a gyárbezárások társadalmi és környezeti hatásaival, végül a leginkább érintett régiókban új üzleti tevékenységek kialakulásának támogatása az európai uniós strukturális és vidékfejlesztési alapokkal összhangban.

Magyarországra nézve egyelőre nehezebben számíthatók ki a reform következményei, ezért a gazdák és a cukorgyártók borúlátóak. Egyes elemzők szerint az öt hazai cukorgyárból 2-3 év múlva legfeljebb három maradhat talpon, és csökkenhet a cukorrépa termőterület és a termelők száma is. Az EU nem ír elő a tagországoknak kötelező kvóta csökkentést, de a fokozatosan mérséklődő felvásárlási árak a gyengébb feldolgozókat, termelőket kiszorítják a piacról.

A megoldást minden piaci résztvevő igyekszik keresni idehaza is, a kialakult helyzet azonban nagyon bonyolult. A hazai cukoriparnak különösen az energiafelhasználás hatékonyságát és a költségtakarékosságot kell növelnie, ugyanis néhány éven belül az olcsó nádcukor árleszorító hatása a répacukor -így a hazai termékek- esetében is érvényesülhet.

4. A CUKORRÉPA BETAKARÍTÁSSAL, TÁROLÁSSAL, SZÁLLÍTÁSSAL KAPCSOLATOS MEGFIGYELÉSEK ÉS JAVASLATOK

A cukorrépa termeltetését, felvásárlását és átvételét a cukorgyárok és a termelők között létrejött szerződések rögzítik. A termelésben az európai uniós szabályokhoz hasonlóan hazánkban is érvényesül a mennyiségi korlátozás. A szakirodalmi munkák alapján megállapítható, hogy a szabályozott és a liberalizálódó cukorpiac képviselői között világméretű

versengés folyik. Ez arra kényszeríti a répacukor gyártó országokat, hogy csökkentsék a cukor előállításának költségeit.

A cukorgyárakban az alapanyag ellátási folyamatok színvonala nagymértékben befolyásolja a cukortermelés eredményességét, ezért azok racionalizálása és korszerűsítése elősegítheti a költségek csökkentését és a versenyképesség megőrzését.

Az átvétel mennyiségi és időbeni ütemezését alapvetően a cukorgyár igényei határozzák meg. A cél az, hogy a megtermelt répa a lehető legkisebb veszteség mellett kerüljön a cukorgyárba. A megfogalmazott cél elérése érdekében a betakarítás, a tárolás és a szállítás folyamatában meg kell keresni a szűk keresztmetszeteket, és ezek ismeretében dönteni arról, hogy hol és milyen változtatásokra, műszaki, valamint technológiai fejlesztésekre van szükség. Ehhez először összefoglaltuk és tanulmányoztuk a külföldön, illetve az itthon már alkalmazott szervezési, irányítási módszereket és átvételi eljárásokat, majd rendszertűztük az alkalmazott szállítási módokat.

4.1. A cukorrépa termeltetés, beszállítás és átvétel hazai és európai gyakorlata

4.1.1. A cukorrépa felvásárlási és átvételi folyamat a Szerencsi Cukorgyár Rt-nél

A hazai termeltetési és átvételi gyakorlatot a Nordzucker AG tulajdonában lévő Szerencsi Cukorgyárban tanulmányoztuk. A Szerencsi Cukorgyár Rt. a megfelelő mennyiségű és minőségű nyersanyag biztosítása érdekében korszerű technikai háttérrel támogatott termeltetési felügyelői hálózatot működtet. A termeltetés folyamatának nyomon követésére, valamint a cukorrépa átvételének és gyártelepre való beszállításának megszervezésére az SAP integrált vállalatirányítási rendszerhez kapcsolódó FP2000 információs és nyilvántartási programrendszer szolgál, amely tartalmazza a teljes termeltetési adatbázist [43].

A cukorrépa betakarítás és átvétel ütemezését a napi feldolgozandó mennyiség ismeretében az FP2000 nyilvántartási rendszerből nyerhető információkra és a korábbi kampányok tapasztalatira támaszkodva tervezik meg.

A cukorrépa táblák pontos helyzetét és méretét, az átvétel helyétől, illetve a gyártól való távolságát műholdas térinformatikai rendszer (GPS) segítségével határozzák meg. A bemért adatok alapján a termelői körzetről digitális térképet készítenek (MapInfo), amely vizuálisan is szemlélteti a táblák elhelyezkedését és méretét, a kijelölt répaátvételi helyeket (depókat), valamint a lehetséges szállítási útvonalakat.

A cukorrépa átvétele a térség kijelölt állomásain, illetve a gyártelepi átvételi központban történik. A gyártelepre érkező cukorrépa szállítmányok mérlegelését 60 t-ás közúti hídmérlegen, illetve az iparvágányra telepített dinamikus vasúti hídmérlegen végzik

Az átvételi központ számítógépes rendszere (FP2000) a betakarítást megelőzően a termelők rendelkezésére bocsátott vonalkód kártya alapján előbb azonosítja a mérlegeléshez érkező szállítmányt (termelő, a cukorrépa származási táblája, szállítóeszköz), majd minden lemért szállítmányról a rögzített adatok alapján mérlegjegyet állít ki, amit átadnak a gépkocsivezetőnek. A mérlegjegy vonalkód és olvasható formában is tartalmazza az átvételre vonatkozó legfontosabb adatokat (a termelő-, a tábla- és a jármű száma, dátum és időpont, bruttó tömeg, mintaszám, ürítőhely).

A felmérés eredményei arra is utalnak, hogy a gyárak a répa beszállítás és átvétel ütemezését, az átvételi helyek (depók) kijelölését döntően gyakorlati tapasztalatokra támaszkodva tervezik meg. Ennek oka vélhetően abban keresendő, hogy a cukorrépa betakarítási és

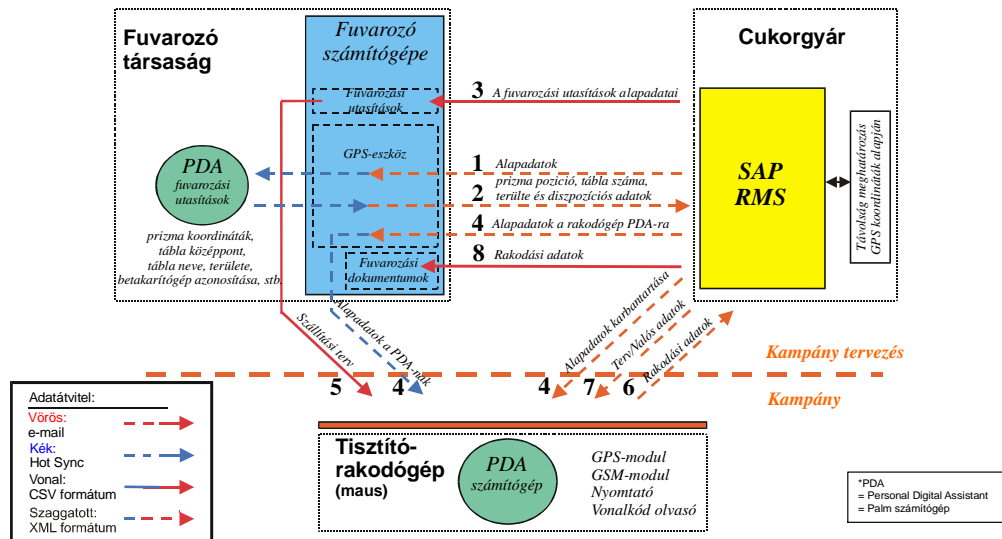
szállítási munkáinak tervezése során nagyon sokféle (biológiai, agrotechnikai, műszaki, gazdaságossági, stb.) szempontot kell figyelembe venni [42].

4.1.2. Az integrált répalogisztikai rendszer felépítése és működése

A cukorrépa betakarítási és szállítási munkák tervezése során nagyon sokféle (biológiai, agrotechnikai, műszaki, gazdasági, stb.) szempontot kell figyelembe venni. Az önmagában is problémát jelentő, hatalmas mennyiségek mozgatását az egyéb körülmények és elvárások tovább nehezítik. A répaszállítást viszonylag rövid idő alatt, nagy területről, több közlekedési alágazat (közút, vasút) és külső átvételi depó munkájának az összehangolásával, a feldolgozási folyamathoz igazodó ütemezéssel és megfelelő minőségben kell megoldani. Ráadásul a tervezést a sztohasztikus hatások (pl. az időjárás) bármikor felboríthatják, ami megköveteli a gyors újratervezés lehetőségét [6].

A vázolt problémák a legkorszerűbb kommunikációs eszközök és a számítástechnika igénybevételével csökkenthetők. Ilyen céllal fejlesztették ki a *MIR* (Modulare Integrierte Rübenlogistik) elnevezésű logisztikai rendszert [35]. A **MIR** betűszóban az **M** (modulált) a rendszerben használt, különböző hardver és szoftver elemek összekapcsolását jelenti. Az **I** (integrált) a gazdák, a szállító-társaságok (speditőrök), a bérvállalkozók (betakarítógép és rakodógép tulajdonosok) és a cukoripari vállalat integrációjára utal. Az **R** jelentése: répalogisztika.

A MIR négy alapfunkciója: a répatáblák, répatároló prizmak koordinátáinak a meghatározása; optimális beszállítási terv készítése számítógépen; online adatforgalom a közreműködők között; a répakiszedés, a rakodás és a beszállítás számítógépes irányítása. A rendszer alkalmazásának előnyei: az emberi tévedések esélyét minimálisra csökkentő automatikus adatrögzítés, a számszerűsített adatok (mennyiségek, költségek, stb.) megbízható, gyors követhetősége.



4.1. ábra. Adatáramlás a MIR rendszerben

A répatáblák koordinátáit, területét és egyéb adatait a cukorgyár központi számítógépén tárolt digitális térképek tartalmazzák, amelyeket ugyancsak *GPS* érzékelővel és *PDA*-val vesznek fel. A megállapodás, illetve a depók helyének kijelölése után a depó koordinátaival, illetve a kiegészítő információkkal frissítik a táblát leíró rekordot, pontosabban a táblához rendelik ezeket az adatokat.

Az objektumok koordinátáinak ismeretében megtervezhető a járművek és tisztító-rakodógépek optimális (legrövidebb úton történő) mozgása. A kinyomtatott térkép egy olyan utasítás, amely rendelkezik a depók felkeresésének sorrendjéről és a depók elérésének útvonalairól.

A cukorgyár, a fuvarozó társaság és a tisztító-rakodó gépet üzemeltető bérvállalkozó közötti online adatáramlást, illetve annak eszközeit (hardver, szoftver) a *4.1. ábra* szemlélteti [10].

A MIR rendszer alkalmazásának előnyei

A répatermesztőnél:

A tábla helyének és területének pontos meghatározása.

A tábla pontos pozíciója automatikusan kerül a répakiszedőre telepített *PDA*-ra.

A táblára vonatkozó hozam adatok és a répa minőségi mutatói lekérdezhetők az interneten.

A betakarításkor gyűjtött adatok felhasználhatók a precíziós répatermelésben.

A fuvarozó társaságnál:

Optimális kampánytervezés, részletes szállítási terv készítése.

A gyors, elektronikus adatközlés lehetővé teszi a szállítás hatékony irányítását.

A digitális térkép (prizmák helye, útvonalak) használata javítja a szállítójárművek navigációs készségét és kihasználását.

A kiszedőgép vezetőjével való kapcsolat révén, gyorsul az információáramlás, és a kiszedett répa mennyiségének ismeretében a szállítókapacitás a tényleges mennyiséghez igazítható.

A répakiszedő-gép vezetőjénél:

Online információ a táblák és a répaprizmák tervezett helyéről.

Egyedi kiszedési terv készítése betakarítógépenként.

Az elszámolás alapjául szolgáló teljesítmények pontos vezetése.

A tisztító-rakodógép vezetőjénél:

Online információ a répaprizmák tervezett helyéről és elérési lehetőségéről.

Aktuális információ a betakarítógép vezetőjétől a kiszedés időpontjáról, a gép állapotról.

A teljesítményre és a rakományokra vonatkozó adatok automatikus rögzítése.

A cukorgyárnál:

A felvásárlási kampány (kiszedés, szállítás és prizmakezelés) optimális tervezése.

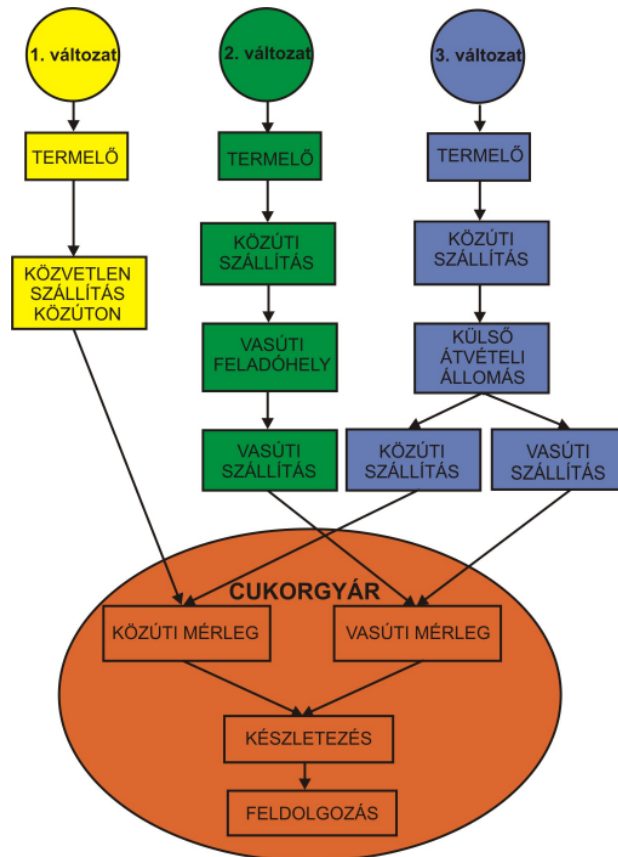
Előzetes információk a beszállításra kerülő répa minőségéről és a várható mennyiségről [10].

4.1.3. A cukorrépa beszállítás módszereinek rendszerezése

A cukorrépa beszállítás módszereit illetően közvetlen (egytagú) vagy összetett (többtagú) szállításról beszélhetünk (*4.2. ábra*). **Közvetlen szállítás** esetén a répa a tábláról, átrakás nélkül, közúti járművek igénybevételel kerül a cukorgyárba, ahol a mennyiségi és minőségi átvételt követően a cukorrépat tisztítják, majd prizmázzák. E megoldás előnyei: a rugalmasság, nem igényli a szállítmány átrakását. Hátrány viszont, hogy nagy mennyiségű szennyező anyag (föld, gyommaradványok, stb.) kerül be a gyár területére, amelynek a szállítása és kezelése növeli a költségeket.

A cukorrépa termelők az egyenletes ütemű betakarításban, a cukorgyárak pedig a termelőkapacitásukhoz igazodó folyamatos feldolgozásban érdekeltek. Ezeknek, a sokszor egymást nem erősítő érdekek érvényesítésének, legjobban az **összetett szállítás** felel meg.

A cukorrépa betakarításban alkalmazott összetett szállításnak ugyanis nemcsak a szállítás folyamatát megszakító szállítóeszköz váltás és átrakás, hanem a betakarítás a feldolgozás kapacitás különbségeit kiegyenlítő pufferolás is lényeges ismérve. A pufferolás, ha nem is teljesen, de korlátozott mértékben képes a termelők és a feldolgozók között feszülő érdekelletet oldani [5].



4.2. ábra. A táblák és a cukorgyár közötti szállítás jelenlegi megoldásai

Az összetett szállításnak az átrakáson és a pufferoláson kívül van még egy nagyon fontos eleme, a tisztítás. A gyárak arra töreksenek, hogy a gyárba szállított répa azonnal feldolgozható legyen, amit ún. fokozott tisztítással lehet elérni. A fokozott tisztításra fejlesztett nagyteljesítményű (150-450 t/h) tisztító-rakodó gépek a cukorrépa táblákon vagy azok közelében a prizmába rakott répát felszedik, tisztítják és szállítójárműre rakják [7].

Az elmondottak alapján könnyen belátható, hogy az eljárás bevezetésének egyik neuralgikus pontja a cukorrépadepók helyének, számának és méretének a megválasztása.

5. SZÁNTÓFÖLDI MÉRÉSEK

5.1. A mérések leírása

A cukorrépa-betakarítással kapcsolatos szántóföldi méréseket Adony és Pusztaszabolcs térségében két különböző nagyságú és alakú táblán végeztük 2005. október 11. és 26. között. A mérések a cukorrépa betakarítással és szállítással összefüggő folyamatok természetének mélyebb megismerése mellett a következő fejezetben tárgyalt szimuláció előkészítésére irányultak. A mérések szántóföld és a vasúti szerelvények megrakása között valamennyi folyamatra kiterjedtek.

5.1.1. A répa kiszedése és prizmázása

A táblákra, a fajtákra, az állományra és a terméshozamra vonatkozó legfontosabb adatokat az 5.1. táblázat foglalja össze.

5.1. táblázat

A szántóföldi mérések fontosabb jellemzői

| | 2005. 11. 11. | 2005.11.26. |
|----------------------------------|---------------|-------------|
| Helyszín: | Besnyő | Adony- |
| Tábla jele | A 22 | C 6 |
| Területe: | 145 ha | 203 ha |
| Fajta: | BOUNTY | KANASTA |
| Sortáv [cm] | 45 | 45 |
| Átlagos tótáv [cm] | 19,2 | 17,8 |
| A hektáronkénti darabszám [db] | 115,6 | 124,5 |
| Egy répa átlagos tömege [kg/db]: | 0,423 | 0,7678 |
| A hektáronkénti hozam [t/ha]: | 48,9 | 95,6 |

Az állomány felméréshez a táblák 10 különböző helyén, 5 m hosszú mérőszakaszokat jelöltünk ki. A szakaszokon mértük a tótávolságot, továbbá a kézzel kiásott, fejezett és tisztított répa tömegét. A mért adatokból számítással határoztuk meg a termésátlagot. A leveles répafej tömege 48,9 t/ha volt.

A szántóföldi mérések mindkét helyszínén a betakarítás Holmer TerraDos KRBS/T2 típusú magajáró fejező-szedő puttonyos (tartályos) betakarítógéppel történt. A gép 7-8 km/h haladási sebesség mellett egy műszakban (12 óra) 10-12 ha répa kiszedésére volt képes. A 24 m³-es tartály telítődési ideje 9-10 perc között változott.

A mérések alkalmával a betakarító gép kiszolgálására egy vagy kettő (attól függően, hogy a betakarító gép csak járműre ürít, vagy közvetlen a prizmára is) JD 6620 típusú, 130 LE-ös traktorral vontatott, Fliegl típusú, tandemkeres, 14 t répa szállítására képes pótkocsi szolgált.

Az egy pótkocsis esetben, a felesleges kiállásokat elkerülendő, a fogások hosszát a termés-hozam figyelembevételével úgy kell kialakítani, hogy a pótkocsi (14 t) és a kombájn tartálya (24 m³ ->18 t) egy hosszban legalább egyszer megteljen, azaz egy fogáshosszról 32 t répát lehessen betakarítani. Így a kombájn egyszer a pótkocsira egyszer pedig közvetlenül a prizmára ürít. Természetesen a fogáshossz úgy is megválasztható, hogy a betakarító gép csak pótkocsira, vagy a közvetlen ürítés mellett több alkalommal pótkocsira ürít <http://www.gek.szie.hu/tanszekek/geti/logisztika/lapok/Cjelentes.htm> .

5.1.2. A répa felszedése, tisztítása, rakodása és szállítása

A tisztító-rakodó gépek használata Magyarországon a '90-es évek elejétől kezdődően fokozatosan terjedt el. Az ezredfordulótól kezdődően pedig a fokozott tisztítást végző magajáró rakodógépek váltak egyeduralmukodóvá, amelyeket a nagy rakodási kapacitás, a kiváló tisztítási hatások és a nagyfokú üzembiztonság jellemez.

A mérések idején a tábla egyik vagy mindkét szélén kialakított prizmában tárolt répát ROPA SRL 15 típusú felszedő, tisztító-rakodógép szedte fel és rakta tehergépkocsira. A répa szállítása a pusztaszabolcsi vasútállomásra Tátra típusú billenhető rakfelületű, terepjáró képességgel rendelkező, pótkocsival ellátott tehergépkocsikkal történt. A magasítókkal szerelt tehergépkocsi szerelvények 22 t répa szállítására alkalmasak, sebességük a terepviszonyoktól függően 40-50 km/h. Az ürítési idő a vasútállomáson 3-5 perc között változott. A két vizsgált cukorrépa tábla közül (5.1 táblázat) az A22 jelű 7 km-re, a C6 jelű 12 km-re feküdt a vasútállomástól. Az első esetben 4 db, a második esetben pedig 6 db tehergépkocsival történt a szállítás.

5.1.3. A répa vagonba rakása

A vasútállomásra szállított, és ott a vasúti pálya mellett (10-14 m-re a sínektől) kialakított, 10-12 m szélességű prizmában tárolt répát a cukorgyári ütemezésnek megfelelően rakták magas oldalfalú nyitott vasúti kocsikba. A rakodásra nagyteljesítményű VOLVO típusú, speciális kanállal szerelt homlokrakodó-gépet használtak. A kanál térfogata egyidejűleg 6 t répa mennyiség emelését és mozgatását teszi lehetővé

<http://www.gek.szie.hu/tanszekek/geti/logisztika/lapok/Cjelentes.htm> .

5.1.4. A fogások hosszának számítása

A fogáshossz és a szükséges járműszám számítására eljárást dolgoztunk ki. A fogások ideális hossza, egyszerű számítással meghatározható, ha ismerjük a hozamot, a sortávolságot, az egy fogáshosszon pótkocsira ürített rakományok számát, a pótkocsival vagy pótkocsikkal szállítható tömeget, a betakarítógép tartályának befogadóképességét és az egyidejűleg kiszedett sorok számát. A fogás hossza, ha a fogás mindkét végén kialakítható prizma :

$$L = \frac{\left(Q + \sum_{i=1}^m Q_i \right) 10000}{(n-1) s H} \quad [\text{m}],$$

ahol:

- Q a betakarítógép tartály befogadó képessége [t],
- Q_i az i -edik pótkocsival szállítható tömeg [t],
- m az egy fogáshosszon pótkocsira ürített rakományok száma,
- n a betakarító gép által egyszerre kiszedett sorok száma [sor],
- s a sortávolság [m],
- H a hozam [t/ha].

Ha csak a fogás egyik végén képezhető prizma, akkor a betakarító gép egy oda-vissza menetben, $2L$ hosszúságú úton egyszer ürít közvetlenül a prizmára és m -szer pótkocsira, ilyenkor az ideális fogáshossz:

$$L = \frac{\left(Q + \sum_{i=1}^m Q_i \right) 10000}{2(n-1) s H} \quad [\text{m}].$$

5.1.5. A pótkocsis szerelvények számának meghatározása

Ürítés két prizmára

Vizsgáljuk a legegyszerűbb esetet, amikor a betakarítógép fogás közben egy alkalommal pótkocsira, a fogás mindkét végén pedig közvetlenül a prizmára ürít. Kérdés egy pótkocsis szerelvény elegendő-e a kiszolgáláshoz.

Például legyen

- a termésátlag: $H = 100$ t/ha,
- a sortávolság: $s = 0,45$ m,
- a kiszedett sorok száma: $n = 6$ sor,
- a pótkocsirakományok száma egy fordulóban $m = 3$,
- a betakarítógép tartályának kapacitása: $Q = 18$ t,
- a pótkocsik kapacitása: $Q_p = 14$ t,
- a betakarítógép sebessége: $v_b = 6$ km/h,
- a pótkocsis szerelvény sebessége: $v_p = 8$ km/h,

a pótkocsis szerelvény ürítési ideje: $t_{ii} = 2,5$ min.

A fogás hossza:

$$L = \frac{(18+14) \times 10000}{(6-1) \times 0,45 \times 100} = 1422 \text{ m.}$$

A betakarítógép tartályának telítődési úthossza:

$$L_b = \frac{Q \cdot 10000}{(n-1) \cdot s \cdot H} = \frac{18 \times 10000}{(6-1) \times 0,45 \times 100} = 800 \text{ m.}$$

A pótkocsis szerelvény mozgása alatt a tartály kétszer telítődik, aminek az időtartama:

$$t_k = \frac{2 L_b}{16,67 v_b} = \frac{2 \times 800}{16,67 \times 6} = 15 \text{ min,}$$

ahol: v_b a betakarítógép sebessége [km/h].

A pótkocsirakomány mennyiséget a betakarítógép

$$L_p = \frac{Q_p \cdot 10000}{(n-1) \cdot s \cdot H} = \frac{14 \times 10000}{5 \times 0,45 \times 100} = 622 \text{ m}$$

úthosszon szedi ki.

Az oda-vissza út megtételéhez szükséges összes idő:

$$t_f = \frac{2L_p}{16,67 v_p} + t_{ii} = \frac{2 \times 622}{16,67 \times 8} + 2,5 = 11,83 \text{ min.}$$

Mivel $t_k < t_f$ a kiszolgáláshoz egy pótkocsi elegendő.

Ürítés egy prizma

Ha egy fogáshosszon több alkalommal ürítünk pótkocsira, és prizma csak a fogás egyik szélén építhető, akkor a pótkocsis szerelvények számának meghatározása kicsit bonyolultabb. A pótkocsis szerelvények száma a következő gondolatmenet alapján számítható.

Az m számú pótkocsirakomány mennyiséget a betakarítógép

$$L_p = \frac{m Q_p \cdot 10000}{(n-1) \cdot s \cdot H} \quad [\text{m}]$$

úthosszon szedi ki, ahol: Q_p a pótkocsirakomány tömege [t].

Az m számú pótkocsirakomány kiszedésének időtartama pedig:

$$t_k = \frac{L_p}{16,67 v_b} \quad [\text{min}],$$

ahol: v_b a betakarítógép sebessége [km/h].

Az m számú ürítési hely távolsága a prizmától a következő számtani sorozat szerint változik:

$$\frac{L_p}{m}, \frac{2L_p}{m}, \frac{3L_p}{m}, \dots, \frac{(m-1)L_p}{m}, \frac{mL_p}{m}.$$

Az ürítési helyek távolságainak az összege:

$$\sum_1^m \frac{L_p}{m} + \frac{2L_p}{m} + \frac{3L_p}{m} + \dots + \frac{(m-1)L_p}{m} + \frac{mL_p}{m} = \frac{L_p}{m} \sum_1^m (1+2+3+\dots+m-1+m).$$

Felhasználva a számtani sor összegére vonatkozó jól ismert formulát az

$$\frac{L_p}{m} m \frac{1+m}{2} = L_p \frac{1+m}{2}.$$

A kiszámított összes távolságot a pótkocsi(k) az oda-vissza menet miatt kétszer teszik meg, így az összes megtett út:

$$L_p(1+m).$$

Az m számú út megtételéhez szükséges fordulódők összege:

$$t_f = \frac{L_p(1+m)}{16,67 v_p} + m t_{ii} \quad [\text{min}],$$

ahol:

v_p a traktoros pótkocsi sebessége [km/h],

t_{ii} a pótkocsi üritési ideje [min].

Végül a **szükséges pótkocsis szerelvéyszám** (z) az egységnyi idő alatt kiszedett és az egységnyi idő alatt szállított répamennyiségek egyenlősége alapján:

$$\frac{m Q_p}{t_k} = z \frac{m Q_p}{t_f},$$

amelyből a

$$z = \text{int} \left(\frac{t_f}{t_k} \right) + 1 \quad [\text{db}].$$

5.2. A megfigyelt és mért adatok

A mérések végrehajtása előtt rendszereztük a megfigyelendő, összegyűjtendő és mérendő adatokat, megterveztük a mérések helyszíneit, elkészítettük a mérési jegyzőkönyvek formátumait, tájékoztattuk a mérésben részvevő kutatókat a feladataikról. Az adatokat és a jegyzőkönyv formátumokat mérőhelyenként rendezve az alábbi pontokban részletezzük.

5.2.1. A répa kiszedés és prizmázás mérése

1. Általános adatok

- 1.1. A mérés időpontja.
- 1.1. A betakarítógép típusa.
- 1.2. Névleges teljesítménye tömeg/idő egységben, sebessége.
- 1.3. A tábla méretei: hossza, szélessége, területe.
- 1.4. Termésátlag.
- 1.5. Szállítójárművek: típusa, száma, teherbírása.
- 1.6. A prizma és a tábla középpontjának távolsága.
- 1.7. Az üritett mennyiség (egy-két szerelvény mérlegelésével)

2. A betakarítógép megfigyelése és mérése

- 2.1. Az események (betakarít, járműre várakozik, ürit, meghibásodás miatt várakozik, egyéb ok miatt várakozik) megfigyelése az idő függvényében (5.3. táblázat).

5.3. táblázat

A betakarítógép megfigyelése

A gép típusa:..... A mérés időpontja:.....

| Sor-szám | Esemény* | Kezdet | Vége | Megjegyzés |
|----------|----------|--------|------|------------|
| | | | | |
| | | | | |

3. A szállítójármű(vek) megfigyelése a betakarítógépnél

- 3.1. A jármű azonosítója (rendszáma vagy egyéb).
- 3.2. A járművel szállított mennyiség (az 1.7. szerint).
- 3.3. A járművel kapcsolatos események a betakarítógépnél: a táblához érkezik, rakodásra várakozik, egyéb ok miatt várakozik, a betakarítógéphez áll, rakodik, a prizmához indul (5.4. táblázat).

5.4. táblázat

A szállítójárművek megfigyelése a betakarítógépnél (betakarítás)

A mérés időpontja:.....

| Sor-szám | Jármű azonosító | Esemény* | Kezdet | Vége | Megjegyzés |
|----------|-----------------|----------|--------|------|------------|
| | | | | | |
| | | | | | |

4. A szállítójárművek megfigyelése a prizmánál

- 4.1. A jármű azonosítója (rendszáma vagy egyéb).
- 4.2. A járművel kapcsolatos események a prizmánál: a prizmához érkezik, ürítésre várakozik, ürít, egyéb ok miatt várakozik, indulás a betakarítógéphez (5.5. táblázat).

5.5. táblázat

A szállítójárművek megfigyelése a prizmánál (betakarítás)

A mérés időpontja:.....

| Sor-szám | Jármű azonosító | Esemény* | Kezdet | Vége | Megjegyzés |
|----------|-----------------|----------|--------|------|------------|
| | | | | | |
| | | | | | |

5.2.2. A répa felszedés, tisztítás, rakodás és szállítás mérése

1. Általános adatok

- 1.1. A mérés időpontja.
- 1.2. A tisztító-rakodó gép típusa.
- 1.3. A tisztító-rakodó gép névleges teljesítménye.
- 1.4. A homlok-rakodógép típusa (a prizmánál).
- 1.5. A homlok-rakodógép rakodási kapacitása [t/h].
- 1.6. A szállítójárművek (vontató és pótkocsi) száma, típusa, teherbírása.
- 1.7. A homlok-rakodógép típusa (a vasúti átadóhelyen).
- 1.8. A homlok-rakodógép (a vasúti átadóhelyen) rakodási kapacitása [t/h].
- 1.9. A prizma és a vasúti átadóhely közötti szállítás távolsága.

2. A tisztító-rakodó és a homlok-rakodó gép megfigyelése

- 2.1. A tisztító-rakodó gép eseményei: működik, járműre várakozik, a rakodógép miatt várakozik, meghibásodás miatt várakozik, egyéb ok miatt várakozik (5.6. táblázat).

5.6. táblázat

A tisztító-rakodó gép megfigyelése

A gép típusa:..... A mérés időpontja:.....

| Sor-szám | Esemény | Kezdet | Vége | Megjegyzés |
|----------|---------|--------|------|------------|
| | | | | |
| | | | | |

(A 2.2. csak a régi technológiában)

- 2.2. A homlok-rakodógép eseményei: a kanalat telíti, teherrel mozog, a kanalat üríti, üresen mozog, tisztító-rakodó gép miatt várakozik, egyéb ok miatt várakozik (5.7. táblázat).

5.7. táblázat

A homlok-rakodó gép megfigyelése a prizmánál

A gép típusa:..... A mérés időpontja:.....

| Sor-szám | Esemény | Kezdet | Vége | Megjegyzés |
|----------|---------|--------|------|------------|
| | | | | |
| | | | | |

3. A szállítójárművek megfigyelése a prizmánál

- 3.1. A jármű azonosítója (rendszáma).
3.2. A járművel szállított mennyiség (az 1.7. szerint szerelvény típusonként).
3.3. A járművel kapcsolatos események a prizmánál: a prizmához érkezik, rakodásra várakozik, egyéb ok miatt várakozik, a tisztító-rakodó géphez áll, rakodik, a vasúti átadóhelyhez indul (5.8. táblázat).

5.8. táblázat

A szállítójárművek megfigyelése a prizmánál (szállítás)

A mérés időpontja:.....

| Sor-szám | Jármű azonosító | Esemény* | Kezde-te | Vége | Megjegyzés |
|----------|-----------------|----------|----------|------|------------|
| | | | | | |
| | | | | | |

4. A szállítójárművek megfigyelése a vasúti átadóhelyen

- 4.1. A jármű azonosítója (rendszáma).
4.2. A járművel kapcsolatos események a prizmánál: a vasúti átadóhelyhez érkezik, ürítésre várakozik, egyéb ok miatt várakozik, ürít, a prizmához indul (5.9. táblázat).

5.9. táblázat

A szállítójárművek megfigyelése a vasúti átadóhelyen

A mérés időpontja:.....

| Sor-szám | Jármű azonosító | Esemény* | Kezde-te | Vége | Megjegyzés |
|----------|-----------------|----------|----------|------|------------|
| | | | | | |

| | | | | | |
|--|--|--|--|--|--|
| | | | | | |
| | | | | | |

5.2.3. A répa vagonba rakás mérése

1. A homlok-rakodó gép megfigyelése a vasúti átadóhelyen

1.1. A homlok-rakodógép eseményei: a kanalat telíti, teherrel mozog, a kanalat üríti, üresen mozog, egyéb ok miatt várakozik (5.10. táblázat).

5.10. táblázat

A homlok-rakodó gép megfigyelése a vasúti átadóhelyen

A gép típusa:..... A mérés időpontja:.....

| Sor-szám | Esemény* | Kezde-te | Vége | Megjegyzés |
|----------|----------|----------|------|------------|
| | | | | |
| | | | | |

5.3. A mérési eredmények feldolgozása

Az 5.3.-5.10. táblázatokban bemutatott mérési jegyzőkönyvekben rögzített egymással összefüggő adatokat a kiszedési, ürítési, rakodási, szállítási, várakozási stb. időközök számlíthatósága érdekében Excel táblákba írtuk. Az Excel táblákban számított adatsorokból meghatároztuk a jellemzők várhatóértékét és szórását, majd az adatokat osztályokba soroltuk, és megrajzoltuk azok hisztogramjait is

<http://www.gek.szie.hu/tanszekek/geti/logisztika/lapok/Cjelentes.htm> .

A mérések legfontosabb adatait és azok statisztikai jellemzőit az 5.11 táblázatban foglaltuk össze.

5.11. táblázat

A mért paraméterek statisztikai jellemzői

| Paraméter | Átlag | Szórás | Minimum | Maximum | Módusz |
|--|-------|--------|---------|---------|--------|
| Betakarítógép kiszedési idő 14 t répa esetén [min] | 5,16 | 1,71 | 2 | 10 | 6 |
| Betakarítógép ürítési idő 14 t répa esetén [min] | 8,58 | 0,51 | 8 | 9,13 | 8,5 |
| A betakarítógép pótkocsira ürít [min] | 4,87 | 2,23 | 2 | 9 | 4 |
| A betakarítógép prizmára ürít [min] | 5,75 | 4,21 | 1 | 13 | 3 |
| A pótkocsi prizmára ürít [min] | 3,86 | 1,07 | 2 | 5 | 4 |
| Pótkocsi menetidő a prizmához [min] | 10,33 | 8,06 | 4 | 22 | 4 |
| ROPA rakodási idő járműre [min] | 6 | 4 | 2 | 16 | 4 |
| Jármű sebesség rakott menet [km/h] | 31,86 | 5,47 | 12,85 | 45 | 32,14 |
| Jármű sebesség üres menet [km/h] | 34,48 | 8,03 | 6,61 | 45 | 34,62 |
| Jármű ciklusidő [min] | 46 | 23 | 33 | 112 | 35 |
| Jármű menetidő rakott | 14 | 3 | 10 | 35 | 14 |

| | | | | | |
|-------------------------------------|----|----|----|----|----|
| [min] | | | | | |
| Jármű menetidő üres [min] | 15 | 13 | 10 | 35 | 14 |
| Jármű ürítés a vasútállomáson [min] | 3 | 0 | 1 | 5 | 4 |
| VOLVO kanál telítés [sec] | 9 | 3 | 3 | 20 | 8 |
| VOLVO rakottan mozog [sec] | 14 | 4 | 5 | 26 | 15 |
| VOLVO kanál ürítés [sec] | 9 | 5 | 4 | 30 | 5 |
| VOLVO üres menet [sec] | 10 | 4 | 4 | 27 | 7 |
| VOLVO ciklusidő [sec] | 49 | 7 | 43 | 66 | 45 |

6. A CUKORRÉPA BETAKARÍTÁS ÉS SZÁLLÍTÁS VIZSGÁLATA SZIMULÁCIÓVAL

6.1. Bevezetés

Ebben a fejezetben a szimulációt a cukorrépa betakarítás modellezésére, az alkalmazott technológia mélyebb megismerésére fogjuk használni. Tekintettel arra, hogy a szimuláció területén használt terminológia korántsem egységes, az alkalmazásfejlesztés megértése és követhetősége érdekében először rövid áttekintést adunk a szimulációs modellekről és nyelvekről, majd bemutatjuk az **Arena** modellezési környezetet, amelyben később elkészítjük a cukorrépa betakarítás és szállítás modelljét. Az alapok ismeretében, lépésről lépésre haladva, ismertetjük a modellfejlesztés fázisait. Végül az üzemi megfigyelések mért adataira alapozva futtatjuk a kifejlesztett modellt és kiértékeljük az eredményeket.

6.2. Szimulációs modellek és nyelvek

A szimulációs modellek a vizsgált rendszer logikai vagy matematikai modelljei, amelyek számos strukturális és kvantitatív közelítést és feltételezést tartalmaznak a valóságos rendszerhez viszonyítva. A logikai modell rendszerint, mint számítógép program jelenik meg, amely a modell viselkedésére vonatkozó kérdésekre ad válaszokat, feltéve, hogy a modell hű reprezentációja a rendszernek. Az utóbbi esetben remélhetjük, hogy új ismereteket szerezhethünk a rendszer viselkedéséről. Mivel a valós rendszer helyett a számítógép programban változtathatjuk a bemenő paramétereket és a modell viselkedését, egy sor kérdésre lényegesen olcsóbb és gyorsabb válaszokat kapunk. Ugyanakkor a kísérletezés során előforduló hibák nem járnak olyan drámai következményekkel, mint amikor azokat a valóságban követjük el. A számítástechnika fejlődése és a számítási költségek csökkenése sok más területhez hasonlóan kedvezően hat a logikai modellek komputer analízisében rejlő lehetőségek kihasználására.

A digitális számítógépek megjelenésével szinte egy időben, az 1950-es és az 1960-as években általános célú programnyelveket, például a **FORTAN**-t használták a szimulációs programok megírására. Ezek a programnyelvek általában a szimulációt támogató csomagokkal is rendelkeztek. A fejlődés következő állomását a speciális célú szimulációs nyelvek jelentették, mint a **GPSS**, **Simsript**, **SLAM**, **SIMAN**, és biztosítottak sokkal jobb, kényelmesebb programozási környezetet a felhasználóknak.

Az utóbbi években jelentek meg az ún. magas szintű szimulátorok, amelyeket elsősorban a könnyű használhatóság, az interaktív grafikus felhasználói felület, a menük és a dialógusok jellemeznek. A szimulátorban a rendelkezésre álló szimulációs modell elemekből választhatunk, azokat összekapcsolhatjuk, paraméterezhetjük és a programot futtatva a rendszerkomponensek mozgását, változását grafikusán megjeleníthetjük (animálhatjuk). Legtöbb szimulátor alkalmazási területe erősen korlátozott (pl. gyártás, kommunikáció), és

általában nem rugalmasak. Sokak szerint ezek a szoftvercsomagok túl messzire mentek azzal, hogy feláldozták a rugalmasságot a könnyű használat oltárán.

A kutatási témánkban alkalmazott **Arena** a magas szintű szimulátorokat jellemző könnyű használatot kombinálja a szimulációs nyelvek rugalmasságával, és emellett még a teljesen általános célú **Microsoft® Visual Basic®** és **C** nyelvek használatát is biztosítja. Teszi ezt oly módon, hogy a grafikus szimulációs modellben alternatív és kicserélhető sablonokról (template-ekről) gondoskodik, és az analizáló modulok is kombinálhatók azért, hogy a szimulációs modellek széles körét korrekt módon építhessük fel.

A hierarchikus felépítésével az **Arena** fenntartja a modellezés rugalmasságát. Az ún. modellablakba bármikor áthúzhatók alacsonyabb szintű modulok a **Blocks** és **Elements** panelekről, és keverhetők a magasabb-szintű modulokkal. Az alacsonyabb szintű modulok rendelkeznek a **SIMAN** szimulációs nyelv adta rugalmassággal. Ugyanakkor írhatunk **Visual Basic** vagy **C/C++** kódú részleteket döntési algoritmusok futtatására, vagy adat-elérési rutinokat külső adatbázisok eléréséhez. Tervezhetünk, létrehozhatunk saját modulokat is (a modulok valójában **SIMAN** komponensekből állnak) és ezeket saját sablonokba gyűjthetjük. Az elmondottak alapján az **Arena** alkalmazásakor nem kell kompromisszumokat kötni a könnyű alkalmazhatóság és a rugalmasság között.

6.3. Az Arena modellezési környezet

Az **Arena** alkalmazásablakban a modellezéskor három területen dolgozunk: a Project Bar-on, a Modellablak folyamatábra nézeten és a Modellablak táblázat nézeten (6.1. ábra). A **Project Bar**-on elérhető, a folyamatok definiálására hivatott **Basic Process**, **Advanced Process** és **Advanced Transfer** panelek tartalmazzák a **moduloknak** nevezett modell formákat (alakzatokat):

- a **Reports** panel lehetővé teszi a szimulációs eredmények kijelzését;

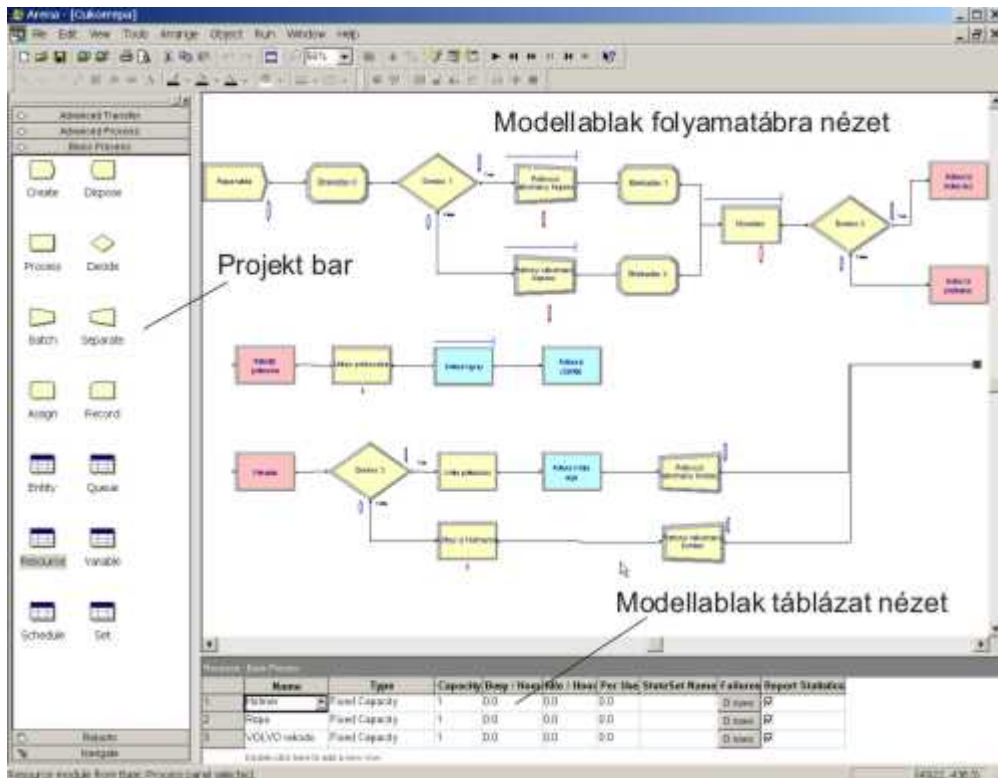
- a **Navigate** panel a modell különböző szempontok szerinti megjelenítését segíti, beleértve a hierarchikusan felépülő almodellek közötti navigációt is.

A **modellablaknak** két fő területe van. A folyamatábra nézet (*flowchart view*) a grafikákat, beleértve a modulokból felépített folyamatábrát, az animációt és egyéb rajzi elemeket tartalmazza. Az alsó ablak a táblázat nézet (*spreadsheet view*) a modell adatokat jelzi ki táblázatosan, mint pl. az idő, a költség és más paraméterek.

A tanulmányban a cukorrépa betakarítást, tisztítást, szállítást és a vagonba rakást modellezzük. Az 6.1 ábra a betakarítási folyamat műveleteit (kiszedés, pótkocsira rakás, szállítás, ürítés pótkocsiról, stb.) ábrázolja a répatábla és a répaprizma között.

A **Project Bar**-on elérhető **modulok** (alakzatok) lehetnek folyamatábra elemek és adatmodulok. A modulok az adott folyamat szimulációhoz szükséges információkat tárolják. Az egymáshoz kapcsolt folyamatábra modulok (pl. *Create*, *Dispose*, *Process*, *Decide*, *Transport*, *Assign*, stb.) határozzák meg, hogy mi történik és milyen sorrendben a szimuláció alatt. E modulok működését befolyásoló paraméterek a modulokhoz rendelt dialógusablakokban adhatók meg. Például a *Process* modul működéséhez meg kell nevezni az erőforrásokat (*Resources*), azok kapacitását, a folyamat időigényét, annak eloszlását stb.

Az **adatmodulok** (*Entity*, *Queue*, *Resource*, *Variable*, stb.) bizonyos objektumhalmazok tulajdonságait leíró modulok, vagyis adatobjektumok, amelyekre a folyamatábra modulok hivatkozhatnak. Például az előzőleg említett *Process* modulban a *Resource* adatmodul valamelyik rekordjára vagy rekordjaira hivatkozunk. Az adatmodulok mezői a modellablak táblázatnézetében jelennek meg és tölthetők fel.



6.1. ábra: Az Arena modellezési környezete a cukorrépa betakarítás folyamatábrájával

A folyamatábrán áramló objektumokat, amely esetünkben a cukorrépa, **entitásoknak** nevezzük. Az entítások változtathatják a státuszukat, hatnak más entításokra és a rendszer állapotára, illetve maguk is más entítások hatásainak lehetnek kitéve. Az entítások dinamikus objektumok, amelyeket rendszerint a szimulációban hozunk létre a *Create* modul segítségével, majd a rendszeren átáramolva a kilépési ponton, ez a *Dispose* modul, elhagyják a rendszert. Léteznek olyan entítások is, amelyek soha nem hagyják el a rendszert, állandó körforgásban vannak. Az entítások másolhatók, csoportosíthatók (batch képzés) és szeparálhatók (batch bontás). Az entítások tulajdonságai az *Entity* adatmodulban adhatók meg.

Az **attribútumok** az entításokhoz kapcsolódnak és egyéni jelleget kölcsönözhetnek azoknak. Az attribútum közös jellemzője egy entitás típusoknak, de specifikus értéket adva alkalmasak az azonos típusú entítások megkülönböztetésére is. Például erre szükség lehet akkor, amikor a FIFO elvet szeretnénk alkalmazni. Egy attribútumot úgy lehet elképzelni, mint egy címkét, amely minden entitáson megtalálható, de a címkén lévő szöveg különbözhet, és egyéni jelleget adhat az entitásnak.

A **global variables** (változók) az egész rendszerre vonatkozó tulajdonságokat jellemeznek tekintet nélkül arra, hogy mennyi és milyen entítások találhatóak a rendszerben. A modellben számos, különböző változó definiálható, de mindegyik egyedi. Az **Arena**-ban kétféle változót különböztetünk meg: beépített változók (pl.: number in queue, number of busy servers, current simulation clock time, stb.) és a felhasználó által definiált változók (pl.: átlagos kiszolgálási idő, utazási idő, pillanatnyi szállítmány tömege, stb.). Az attribútumokkal ellentétben a változók nem kapcsolódnak egyedi entításokhoz, hanem az egész rendszerre vonatkoznak. A változók az entítások számára elérhetők, és sok esetben az entítások képesek a változók értékét megváltoztatni.

A változók sok- és különféle célra használhatók. Például két állomás közötti szállítási idő a modellben lehet mindenütt ugyanakkora, ilyenkor célszerű a *Szállítási idő* nevű változót definiálni. Az **Arena** változók lehetnek vektorok, mátrixok, és így kényelmesen listákba

vagy két dimenziós táblázatokba rendezhetők. Itt jegyezzük meg, hogy az **Arena** változók, attribútumok, modulnevek stb. nem tartalmazhatnak ékezetes karaktereket. Ez a megszorítás az oka annak, hogy a magyar szavakat az általunk fejlesztett modellben ékezet nélkül írjuk.

Az entitások kiszolgálását végző objektumok a **resources** (erőforrások), amelyek lehetnek emberek, eszközök, tárolóhelyek, stb. korlátozott méretekkkel. Az erőforrások igénybevétele jelentő lehetséges tevékenységek (actions): **seize** (megfogás), **delay** (késleltetés) és **release** (elengedés). Ezek a tevékenységek késedelem nélkül, egymást követően, vagy időeltolódással, valamilyen feltételhez kötötten jelentkezhetnek. Az erőforrásokat úgy kell elképzelni, hogy azok az entitásokhoz rendelve és nem fordítva, mivel egy entitást egyidejűleg több erőforrás is kiszolgálhat. Egy erőforrás több egyedi egységből (unit of resource) álló csoportot is reprezentálhat. Ez hasznos lehet akkor, amikor azonos egységek működnek párhuzamosan, például egy áruházban a pénztárak. Az egységek száma a szimuláció futása alatt változtatható, például a sorok hosszától függően pénztárakat nyithatunk meg és zárhatunk be.

Amikor egy entitás a modellben megadott szekvenciának megfelelően nem tud tovább mozogni, mert a soron következő művelet erőforrása foglalt, akkor várakozásra kényszerül. A várakozások helyei a **Queues** (sorok). A sorok az **Arena**-ban saját névvel rendelkeznek, megadható a kapacitásuk, például korlátozott kapacitású tároló, továbbá megadható a sor kiszolgálásának elve (típusa), ami lehet FIFO, LIFO, stb.

A szimuláció eredményeinek követését szolgálják az ún. **Statistical Accumulators**-nak (statisztikai akkumulátoroknak) nevezett változók, amelyek az entitások, a sorok, az erőforrások, folyamatok, állomások, stb. statisztikai jellemzőiről gyűjtenek adatokat. Például az adott időpontig létrehozott entitások számáról, az összes várakozási időről, az adott soron átmenő entitások számáról, adott sorban eltöltött leghosszabb időről, stb.

Alapvetően a szimuláció során minden az **events** (események) körül forog. Az esemény azt jelenti, hogy egy adott időpontban történik (érkezik, befejeződik, távozik, stb.) valami, ami megváltoztathatja az attribútumok, a változók és a statisztikai akkumulátorok értékét. Az események követhetőségét az ún. **event calendar** teszi lehetővé, amelyben az **Arena** az eseményekre vonatkozó információkat tárolja. Egy esemény rekord tartalmazza az entitás azonosítóját, az esemény időpontját és magát az eseményt. A kalendárium tetején mindig a legutolsó esemény helyezkedik el.

6.4. Modellépítés az Arena környezetben

Az **Arena Professional Edition** és az **Arena Standard** változata együttesen egy teljesen integrált fejlesztési környezetről gondoskodik, ami lehetővé teszi, hogy grafikus animációs modelleket építsünk, továbbá ellenőrizzük és analizáljuk azokat. Ez jelenti azokat a moduloknak nevezett, újrahasználatos modellezési komponenseket, amelyeket könyvtárakba vagy sablonokba gyűjthetünk.

6.4.1. Elemek (elements) és tulajdonságok (properties)

Amikor egy modult helyezünk egy modellbe, annak jellemzői az adott másolatban specifikusak, az operanduszok értékének vagy az objektum megjelenésének a megváltozása a *user view*-ban azonban nincs hatással más azonos típusú modulmásolatokra.

Egy szimulációs modellben azonban akad néhány építőelem, amelyek természetüknél fogva globálisak és örökletesek. Ezeket a modell elemeinek **elements** nevezzük, és ezekre az elemekre nemcsak egy, hanem több modulmásolat is hivatkozhat. Egy elem létrehozása-

kor, az elem neve felkerül egy listára, ahonnan elérhetővé válik más modulmásolatok számára is.

Az elemek típusokba csoportosítottak, mint pl. *queues* (sor), *resources* (erőforrás), *conveyors* (konvektor), stb. Minden csoport sajátos karakterisztikákkal rendelkezik, amelyeket *properties*-nek (tulajdonságoknak) nevezünk. A sor tulajdonsága például a sorrend szabály, az erőforrásokhoz kapacitás, meghibásodás, stb. rendelhető, a konvektor tulajdonsága a sebesség, továbbá a konvektor lehet akkumuláló vagy nem-akkumuláló típusú.

A modellben definiált specifikus elemek tulajdonságai saját értékekkel rendelkeznek. Fontos megjegyezni, hogy az egyedi elemek tulajdonságai globálisak a teljes szimulációs modellben. Ha például egy modell egyik moduljában inicializálunk egy erőforrást és hozzárendeljük a tulajdonságait, akkor egy később a modellbe helyezett *Resource* modulban (ugyancsak a **Common** panelről) erőforrást választva az erőforrás listából, ugyanazok a tulajdonságok jelennek meg, amelyeket a korábban már megadtunk.

Egy modul definícióban egyedi operanduszokat azonosíthatunk, amelyek specifikus *operand type* (operandusz típusú) elemeket definiálnak. Hasonlóan egy operandusz, amely definiálja egy elem tulajdonságát *Property type*-ként adott.

6.4.2. Folyamatábra és adatmodulok

Bár minden modul sok közös jellemzővel rendelkezik, néha mégis hasznos a sablon tervezéskor különbséget tenni folyamatábra (*flowchart*) és adat- (*data*) modulok között. A *flowchart* modulnak a terminológia azokat a modulokat nevezi, amelyekbe az entitások beáramlanak, illetve amelyekből kiáramlanak. Ilyenek az **Arena Basic Process** panelen a *Create*, *Dispose*, *Process*, *Decide*, *Batch*, *Separate*, *Assign* és *Record* modulok. Ezek alapvető termelési modulok, amelyek az entitásokra hatnak.

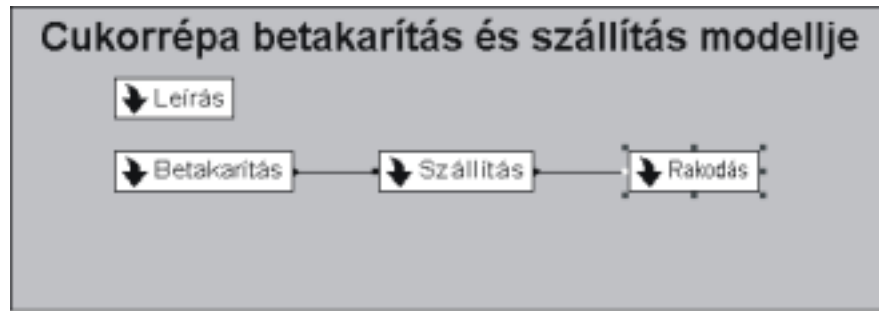
Ezzel szemben a *data* (adat) modulokon az entitások nem folynak keresztül. A *data* modulok a szimulációs modell elemeiről szolgáltatnak információkat. Az **Arena Basic Process** panel *data* moduljai: *Entity*, *Queue*, *Resource*, *Variable*, *Schedule* és *Set*.

Amíg a *flowchart* modulokat a **Projekt** panelről áthúzzuk (áthelyezzük) modellablakba és az alakzatokat folyamatábrává kapcsoljuk össze, hogy leírjuk a rendszer logikai folyamatát, addig a *data* modulokat nem kell a modellablakba helyezni. A *data* modulok kiválasztás után a modellablak *spreadsheet* (táblázat) nézetében szerkeszthetők.

6.5. A cukorrépa betakarítás logikai modellje és paraméterei

A kutatás előzményeire és az 5. fejezetben ismertetett mérések eredményeire támaszkodva, napjaink legkorszerűbb cukorrépa betakarítási technológiáját modelleztük, amelynek vezérgépei a nagyteljesítményű (1,5 ha/h) puttonyos betakarítógép és fokozott tisztítást végző felszedő rakodó gép. A modellezés során szerzett tapasztalatok és módszerek természetesen átültethetőek más, a vizsgálnál kevésbé korszerű betakarítási technológiákra is.

A szimulációs vizsgálat a kiszedés és a vasúti szerelvény megrakása közötti műveleteket foglalja magában és célja a rendszerelemek kapacitásainak az összehangolása. A műveletek három almodellbe csoportosíthatók, és ezek akár külön is modellezhetők. E műveletek: a répa kiszedése; a betakarítógép üritése pótkocsira vagy prizmára; pótkocsis szállítás a betakarítógép és a prizma között; a pótkocsi üritése prizmára; felszedés, tisztítás és rakodás tehérgépkocsira felszedő-tisztító rakodó géppel; szállítás a prizma-vasútállomás útvonalon; a tehérgépkocsi üritése a vasútállomáson; a vasúti szerelvény megrakása (kanál telítés, rakott menet, kanál ürités és üresmenet) kanalas homlokrakodó géppel.



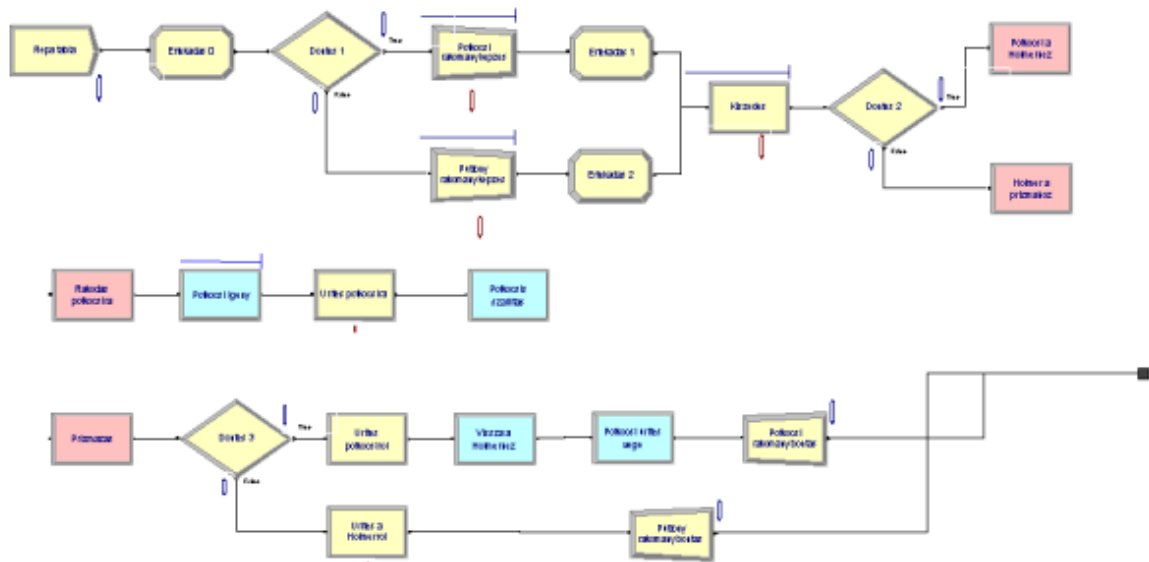
6.2. ábra: A cukorrépa betakarítási és szállítási modell belépési pontja

Az almodellek határai pufferek, esetünkben a prizmák, ahol elméletileg korlátlan hosszúságú sorok is keletkezhetnek. Ezek a sorok jelentik a következő művelet sor forrását. Az első csoportba a kiszedés és a szántóföldi prizmázás közé eső műveletek, a második csoportba a szántóföldi prizma és a vasútállomás között elvégzendő műveletek, a harmadik csoportba pedig a vasúti szerelvény megrakása sorolhatók.

Az almodellek elágazási helye a menüszerűen működő ún. **Top-Level Modell** (6.2. ábra), ahol a megfelelő fülre kattintva megtekinthetjük az almodellek logikai hálóját. A továbbiakban az almodellek fejlesztésének lépéseit részletesen bemutatjuk.

6.5.1. A betakarítás almodellje

A betakarítási almodell logikai hálóját a 6.2. ábra szemlélteti, amelyen az almodellbe szerkesztett *flowchart* modulok és a köztük lévő kapcsolatok láthatók. A modellablakba helyezett modulok a **Basic Process**, az **Advanced Process** és az **Advanced Transfer** panelekhez tartoznak, amelyeket korábban a modellhez csatoltunk a **File->Template panel->Attach** menüponttal.



6.2. ábra: A betakarítási almodell

A modell belépési pontja a „Repa tabla” nevű *Create* modul. A *Create* modul feladata az entitások létrehozása. A modulban létrehozott entitások 1 t mennyiségű cukorrépanak felelnek meg, és ezek állandó, 0,6 min/entitás intenzitással keletkeznek, amit a betakarítógép mért teljesítmény adatai alapján határoztunk meg úgy, hogy a modellbe áramló répa mennyiség biztosítsa a betakarítógép várakozás nélküli, folyamatos működését.

Az „Ertekadas 0” nevű *Assign* modul dialógusában entitás attribútumokat definiálunk a későbbi batchképzéshez. (Egy *Assign* modulban új értéket adhatunk a változóknak, az entitások attribútumoknak, az entitás típusoknak, az entitást megjelenítő képeknek és más rendszerváltozóknak.) Az első attribútum neve „Potkocsi rakomány”, értéke pedig 14, ami a betakarítógépet kiszolgáló pótkocsi 14 t-ás kapacitásának felel meg.

A „Dontes 1” nevű *Decide* modulban arról döntünk, hogy a kiszedett répa pótkocsira lesz ürítve vagy közvetlen a prizmára. Ez egy két utas, feltételhez kötött döntés, amelyhez a „Jarmure” nevű változót használjuk, amely 0 és 1 értéket vehet fel, és kezdeti értéke 1. A *Jarmure* =0 azt jelenti, hogy a puttony térfogatának megfelelő 18 t tömegű batch-et, a *Jarmure* =1 pedig azt, hogy a pótkocsi térfogatának megfelelő 14 t-ás rakományt kell képezni.

A rakományokat a „Potkocsi rakomány kepezés” és a „Puttony rakomány kepezés” nevű *Batch* modulokban hozzuk létre. (A *Batch* modul a csoportosító mechanizmus szerepét tölti be a szimulációs modellben. A batch-ek lehetnek állandóak vagy ideiglenesek. Az ideiglenes batch-ek a később a *Separate* modullal oszthatók meg.) A „Potkocsi rakomány kepezés” dialógusában batch típusa ideiglenes, azaz a *Type* listában *Temporary*-t választunk, mivel később visszatérünk az eredeti, 1 t-ás entitáshoz, a *Batch Size* mezőbe beírjuk a „Potkocsi rakomány” attribútumot, amit korábban az „Ertekadas 0” nevű *Assign* modulban definiáltunk.

Betakarításkor a fogások hosszát a tábla méretétől és a termésmennyiségtől függően úgy választják meg, hogy a betakarítógép puttonya éppen a fogás végén teljen meg (ekkor a betakarítógép közvetlen a fogás két végén kialakított prizmára ürít). A fogás hossza úgy is megválasztható, hogy arról egy pótkocsi rakomány (14 t) és egy puttony rakomány (18 t) mennyiségű répát lehessen betakarítani. Ebben az esetben a betakarítógép egyszer pótkocsira, egyszer pedig közvetlen a prizmára ürít. Az utóbbi esetben a betakarítás során a két rakományméret ciklikusan váltogatja egymást, amit a modellben úgy érünk el, hogy az „Ertekadas 1” és az „Ertekadas 2” nevű *Assign* modulokban ciklikusan változtatjuk a „Jarmure” változó értékét.

A „Potkocsi rakomány kepezés” nevű *Batch* modult követő „Ertekadas 1” nevű *Assign* modul dialógusa látható, amelyben a „Jarmure” változó értékét 2-re változtatjuk, azaz a következő, újonnan létrehozott entitásokat a *Decide* modulból „Puttony rakomány kepezés” nevű *Batch* modulba küldjük.

Az „Ertekadas 1” nevű *Assign* modulban további attribútumokat („Szallitasi mod” és „Kiszedési ido”) definiálunk. Az *Add* gombra kattintva megnyílik az *Assignments* ablak (6.26. ábra), amelyben megadhatjuk az attribútumok értékeit. A „Szallitasi mod” attribútum értéke alapján döntünk arról, hogy az ürítés pótkocsira („Szallitasi mod” = 1), vagy közvetlen a prizmára („Szallitasi mod” = 2) történjen.

A „Kiszedési ido” a mérések alapján trianguláris eloszlású valószínűségi változó, 14 t-ás pótkocsi rakomány esetén az eloszlás paraméterei: TRIA(6,7,10), ahol az első paraméter a legkisebb, a második a legvalószínűbb és a harmadik a legnagyobb időérték.

A modulban megadjuk a batchképzéssel létrehozott új entitás típusát (Repa 14 t) és az *Entity Picture* értékét *Picture Blue Ball*-ra változtatjuk, vagyis a szimuláció ideje alatt a kék labdák fogják a 14 t-ás rakományt reprezentálni.

Az elmondottakból következően a *Kiszedes* nevű *Process* modulba belépő entitások 14 t -ás pótkocsi rakományok vagy 18 t -ás puttonyrakományok, azaz a továbbiakban kétféle és egyedi tulajdonságokkal felruházott entitással dolgozunk. Az attribútumok („Kiszedési ido” és a „Szallitasi mod”) az entitások mérete és szállítási módja miatt különbözőek. A

többször is használt „Szallitasi mod” attribútum biztosítja, hogy a répát a rakomány méreteinek megfelelő műveletsorra irányítsuk.

A „Kiszedes” nevű *Process* modul dialógusában megadjuk a kiszedés erőforrását, a betakarítógépet, a modulhoz rendeljük a rakomány méretétől függő „Kiszedesi ido” attribútumot és a *Seize Delay* tevékenységet. Az utóbbit az *Action* lenyíló listából választjuk. A kiszedési idő megadásához a *Delay Type* listában az *Expression*, a *Units* listában pedig a *Minutes* opciókat jelöljük meg. Ezután az *Expression* mezőbe írjuk be a „Kiszedesi ido” attribútumot.

A répakiszedés erőforrása a betakarítógép. A *Release* action-t, azaz a betakarítógép felszabadítását később a rakodás befejezésekor az „Urites potkocsira” és az „Urites a Holmerrol” nevű *Process* modulokban hajtjuk végre. Ez azt jelenti, hogy a betakarítógép csak akkor képes új entitást fogadni, ha az ürítés már befejeződött. Az esetlegesen feldolgozásra váró entitások a *Process* modulhoz tartozó sorban várakoznak.

A kétféle entitásnak és szállítási módnak megfelelő elágaztatás a „Dontes 2” modulban történik meg, ahonnan a „Szallitasi mod” attribútum értékétől függően a „Potkocsi a Holmerhez” vagy a „Holmer a prizmahoz” nevű *Route* modulokra irányítjuk az entitásokat.

Az **Advanced Transfer** panelen található *Route* modulban adható meg a célállomás és a szállítási idő. A pótkocsis szállítás ágán a „Potkocsi a Holmerhez” nevű *Route* modulban a célállomás típusa: *Station* és a célállomás neve: „Rakodohely”, amit a „Rakodas potkocsira” nevű *Station* modulban definiálunk. A *Route Time* mező értéke azért nulla, mert az ürítési és a szállítási időt máshol adjuk meg.

A szállítás másik ágán, amikor a betakarítógép közvetlen ürít, a puttony rakomány a „Holmer a prizmahoz” nevű *Route* modulról közvetlen a „Prizmazas” nevű *Station* modulra kerül.

A továbbiakban kövessük a „Rakodas potkocsira” nevű *Station* modulra irányított pótkocsi rakomány útját. Az entitás az állomást elhagyva belép a „Potkocsi igény” nevű *Request* modulba. E modul feladata az **Advanced Transfer** panelhez tartozó *Transporter* nevű *data* modulban definiált szállítóeszközök valamelyikének az igénylése. A táblázatosan megjelenő *data* modulban a következő tulajdonságok (*properties*) találhatóak: *Name* (név), *Capacity* (kapacitás), *Distance Set* (távolság halmaz), *Velocity* (sebesség), *Units* (időegység), *Initial Positions* (kezdő pozíció). A *Transporter Name* lenyíló listából a „Potkocsi” nevű eszközt, a *Selection Rule* lenyíló listából a „Cyclical” opciót választjuk. Megadjuk a pótkocsi sebességét üres menetben, ami szintén trianguláris eloszlású: TRIA(120, 130, 150) m/min.

A pótkocsi által megtett utat az **Advanced Transfer** panelen található *Distance* nevű *data* modulban definiáljuk *Potkocsi.Distance* néven, mint a „Rakodohely” és a „Prizma” nevű állomások közötti távolságot. A *Distance* és a *Transporter* adatmodulok közötti kapcsolatot úgy hozzuk létre, hogy a *Potkocsi.Distance* tulajdonságot *Transporter data* modul *Distance Set* tulajdonságához rendeljük. A sebességből és a távolságból a szimulációs program számítja a szállítás időszükségletét. A pótkocsira várakozó entitások a *Potkocsi igény.Queue* nevű sorba kerülnek.

A pótkocsi ürítést az „Urites potkocsira” nevű *Process* modullal szimuláljuk, amelynek *Action* mezejéhez a *Delay Release* aktivitást rendeljük. Az ürítés értelem szerűen a Holmer betakarítógépet köti le, és az igénybevétel időtartama az ürítési idővel egyenlő. Az ürítési idő eloszlása és paraméterei: TRIA(1, 1.15, 2) min. Az ürítés befejezésekor a *Release action* felszabadítja a betakarítógépet, és az ismét készen áll az új entitás fogadására.

A „Pótkocsis szállítás” nevű *Transport* modul dialógusablakában adjuk a *Transporter* adatmodulban definiált szállítóeszközt, a pótkocsit, annak sebességét és a *Distance* adatmodulban definiált szállítási távolságot. A pótkocsis szállítás célállomása a „Prizma” nevű állomás, amit a „Prizmazas” nevű *Station* modulban definiáltunk.

Az eddig leírtak alapján a a „Prizmazas” nevű *Station* modulba két modulból áramlanak entitások, a „Holmer a prizmahoz” nevű *Route* modulból és a „Pótkocsis szállítás” nevű *Transport* modulból. A folyamatban ezért, a „Prizmazas” modult követően, a *Dontes 3* modulban a „Szallitasi mod” attribútumot használva ismét elágaztatjuk az entitásokat.

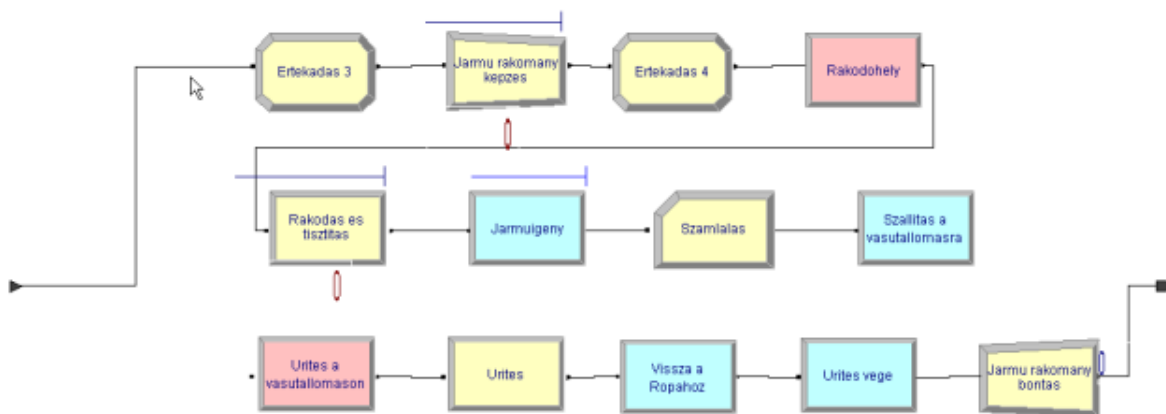
A pótkocsirakományokat „Urites potkocsirol” nevű *Delay* modulban ürtjük, majd a „Vissza a Holmerhez” nevű *Move* modulból a „Rakodohely” nevű állomásra irányítjuk a pótkocsit, végül a „Potkocsi urites vege” nevű *Free* modulban felszabadítjuk a pótkocsit.

A puttonyrakományokat az „Urites a Holmerrol” nevű *Process* modulban ürtjük. A *Process* modulhoz rendelt tevékenységek *Delay Release*, azaz ebben a modulban tesszük szabaddá a betakarítógépet az új entitás fogadására.

Végül a „Potkocsi rakomány bontás” és a „Puttony rakomány bontás” nevű *Separate* modulokban visszaállítjuk az eredeti „Repa 1 t” nevű entitásokat. A betakarítási almodell fejlesztése ezen a ponton fejeződik be.

6.5.2. A szállítás almodellje

A tisztítás, rakodás és szállítás műveleteit tartalmazó **Szállítás** nevű almodell folyamat-ábrája a 6.3. ábrán látható. A betakarítási almodellből a szállítási almodellbe belépő entitások 1 t tömegűek, amelyekből a „Jarmu rakomány kepzés” nevű *Batch* modulban a járművek teherbírásának megfelelő 22 t-ás *Repa 22 t* nevű új entitásokat kreálunk. Ehhez az „Ertekadas 3” nevű *Assign* modul dialógusában definiáljuk a „Jarmu rakomány” nevű attribútumot, aminek 22 t értéket adunk.



6.3. ábra: A szállítás almodellje

Az entitások típusát az „Ertekadas 4” nevű *Assign* modulban „Repa 22 t”-ra változtatjuk, és ehhez az entitáshoz hozzárendeljük a „Picture.Truck” nevű képet, így a szimuláció alatt a 22 t-ás rakományokat egy tehergépkocsi szimbolizálja.

Az új entitások a „Rakodohely” nevű *Station* modulba kerülnek. E modulra csak azért van szükség, hogy a *Distance* adatmodulban definiálhassuk a rakodási pont és a vasútállomás közötti távolságot. Az állomásnak a „Ropa allomas” nevet adjuk.

A répa felszedés, tisztítás és rakodás műveletit a „Rakodas es tisztitas” nevű *Process* modul szimulálja, amelyben erőforrásként a *Resource* adatmodulban definiált „Ropa” tisztító-rakodó gépet adjuk meg a paramétereivel együtt, a műveleti idő eloszlása és paramétere: TRIA(4,5,7).

A „Jarmu” nevű szállítójárműveket, azok sebességét, számát a *Transporter* adatmodulban adjuk meg. A szállító járműveket a „Jarmuigeny” nevű *Request* modulban kötjük le, és a szállítást a „Szallitas a vasutallomasra” nevű *Transport* modullal realizáljuk. A *Request* és a *Transport* modulokban szállítóeszközként a *Transporter* adatmodulban specifikált „Jarmu” nevű eszközt jelöljük meg. A járművek sebessége trianguláris eloszlású: TRIA(36,44,56) km/h.

Az „Urites a vasutallomason” nevű *Station* modul dialógusában az állomásnak a „Vasutallomas” nevet adjuk. Ezt az állomást jelöltük meg célállomásként a „Szallitas a vasutallomasra” *Transport* modulban, továbbá a *Distance* adatmodulban a „Ropa allomas” és a „Vasutallomas” pontok közötti távolságot definiáljuk szállítási távolsággként.

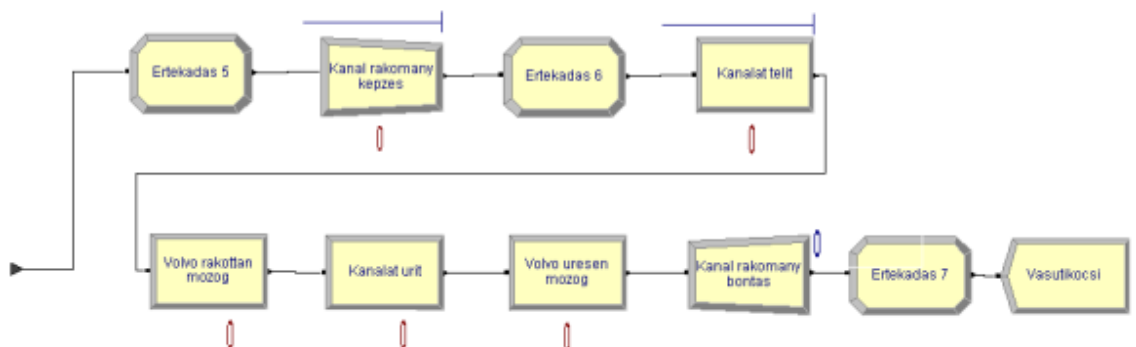
Az „Urites a vasutallomason” nevű *Station* modulba belépő járműveket az „Urites” nevű *Delay* modulban ürítjük. A modulban a trianguláris eloszlású *Delay Time* TRIA(2,3,4,6) min reprezentálja az ürítési időt.

A „Vissza a Ropához” nevű *Move* modulban az üres járműveket visszairányítja a kezdőpontra, „Ropa allomas”-ra, majd az „Urites vege” nevű *Free* modulban felszabadítjuk a járművet.

A folyamat végén a 22 t-ás entitásokat a „Jarmu rakomany bontas” nevű *Separate* modulban ismét 1 t-ás entitásokká alakítjuk.

6.5.3. A rakodás almodellje

A vasúti szerelvény megrakásának logikai almodelljét az 6.4. ábrán vizsgáljuk.



6.4. ábra: A vasúti szerelvény megrakásának almodellje

Az „Ertekadas 5” *Assign* modulban attribútumként adjuk meg a rakodógép kanalának kapacitását, majd a „Kanal rakomany kepzes” *Batch* modulban új, a kanal kapacitásával megegyező 6 t-ás entításokat hozunk létre és az „Ertekadas 6” *Assign* modulban „Repa 6 t” nevet adunk az entításoknak.

Az új entitás belép a „Kanalat telit” *Process* modulba, amelynek erőforrása az *Resource* adatmodulban megadott „VOLVO rakodo”. Az erőforrás hozzárendeléshez kattintsuk az *Add* gombra és a megnyíló *Resources* ablakban (6.61. ábra) definiáljuk a „VOLVO rakodo”-t.

A kanáltelítés művelethez rendelt tevékenységek *Seize Delay*, ami azt jelenti, hogy lekötjük a rakodógépet és a kanáltelítés időtartamáig működtetjük, de felszabadítani csak később, a kanálürítés után fogjuk. A kanáltelítés időtartama trianguláris eloszlású valószínűségi változó TRIA(3,9,20) sec. Az *Allocation* listán ezt az időtartamot *Transfer* időként adjuk meg.

A következő műveletek a rakott menet és a kanál ürítés, amelyeket a „Volvo rakottan mozog” és a „Kanalat urit” *Process* modulok reprezentálnak. Ezekben a modulokban az erőforrás az előző modulban lekötött „VOLVO rakodo”, a tevékenység pedig *Delay*, azaz a rakott menet és az ürítés ideje: TRIA(5,14,26) és TRIA(4,8,30) sec.

A rakodási ciklus utolsó elemét, az üres menetet a „Volvo uresen mozog” *Process* modul szimulálja. A modul erőforrása a „VOLVO rakodo”. Ebben a modulban szabadítjuk fel a rakodógépet, ezért *Delay Release* tevékenységet adunk meg. Az üres menet időtartama: TRIA(4,10,27).

A „Kanal rakomany bontas” nevű *Separate* modulban ismét visszaállítjuk az eredeti entitásméretet és ennek a típusát „Repa 1t” Az „Ertekadas 7” nevű *Assign* modul dialógusában adjuk meg.

Az entitás (a cukorrépa) mozgásának végpontja a „Vasutikocsi” nevű *Dispose* modul. Az entitásokról szeretnénk statisztikát készíteni, ezért a *Record Entity Statistics* kapcsolót bekapcsoljuk. (A modellben *Dispose* modul jelzi a szimuláció végét.)

Végül bemutatjuk a modell fejlesztése során a moduldefiníciókkal párhuzamosan és automatikusan keletkező, illetve az önállóan létrehozott adatmodulokat, amelyek a modellablak táblázatos nézetében jelennek meg.

A modellben kilenc helyen képződhetnek sorok, ahol az entitások várakozásra kényszerülhetnek. A *Queue* adatmodulban többek között az entitások kiválasztásának szabályát változtathatjuk. Az adatmodul tulajdonságai (*properties*):

Name (a sor neve),

Type (az entitások kiválasztásának szabálya lehet: First In First Out, Last In First Out, LIFO, Lowest Attribute Value First, Highest Attribute Value First)

Shared (bekapcsolt állapotban jelzi, hogy a sor a modellben több helyen is használt)

Report Statistics (bekapcsolt állapotban statisztika gyűjtése a sorról)

A *Resource* adatmodulhoz tartozó táblázatban a járművek kivételével a betakarítás és rakodás erőforrásai találhatóak. Az adatmodul tulajdonságai:

Name (az erőforrás neve), *Type* (az erőforrás kapacitás meghatározásának módja, ami lehet állandó vagy az ütemezéstől függő),

Capacity (az erőforrások száma), *Busy/Hour* (a produktív működési idő költsége),

Idle/Hour (az improduktív idő költsége),

Per Use (a teljes használati időre eső költség),

SateSet Name (az erőforrás állapotainak a neve),

Failures (a meghibásodások listája),

Report Statistics (bekapcsolt állapotban statisztika gyűjtése az erőforrásról).

Az *Entity* adatmodul tulajdonságai:

Entity Type (az entitás típus neve, a névnek egyedinek kell lennie),

Initial Picture (az entitást jellemző kezdő kép),

Holding Cost/Hour (a műveletek egy órára eső összes költsége a szimuláció alatt a belépés és kilépés között),

Initial VA Cost (a produktív idő fajlagos költsége, az összes produktív tevékenységnél felmerül),
Initial NVA Cost (az improduktív idő fajlagos költsége, az összes improduktív tevékenységnél felmerül),
Initial Waiting Cost (a fajlagos várakozási költség),
Initial Transfer Cost (a fajlagos szállítási költség),
Initial Other Cost (a fajlagos egyéb költség),
Report Statistics (bekapcsolt állapotban statisztika gyűjtése az entitásról).

A **Transporter** adatmodul tulajdonságai:

Name (a szállítóeszköz neve, ami olyan szállítóeszköz halmazt is reprezentálhat, amelynek minden eleme egymástól független mozgást végezhet),
Capacity (az egymástól független szállítóeszközök száma a szállítóeszköz halmazban),
Distance Set (a *Distance* adatmodulban definiált távolsághalmaz neve, amelynek elemei az állomások és az állomások közötti távolságok),
Velocity (a szállítóeszköz halmaz elemeinek kezdeti sebessége távolságegység /idő-egység mértkegységben adva),
Units (az idő mértékegysége),
Initial Positions (a szállítóeszköz halmaz elemeinek külön-külön megadható kezdő helyzete),
Initial Position (a szállítóeszköz kezdő helyzete, az alapértelmezett érték egy önkényesen választott állomás, a specifikus érték a felhasználó által választott állomás),
Station Name (a szállítóeszköz kezdeti helyzetét meghatározó állomás neve),
Report Statistics (bekapcsolt állapotban statisztika gyűjtése a szállítóeszköztől).

A **Distance** adatmodul tulajdonságai:

Name (a távolsághalmaz neve),
Beginning Station (a start állomás neve),
Ending Station (a végállomás neve),
Distance (a távolság).

6.6. A cukorrépa betakarítási modell futtatása

6.6.1. A szimuláció paramétereinek beállítása

Mielőtt futtatnánk a modellt, először szintaktikai szempontból ellenőrizzük a működését. Ezt megtehetjük úgy, hogy a **Run Interaction** eszközsoron a **Check** gombra (√) kattintunk, vagy a **Run>Check Model** menüpontot választjuk, amit a klaviatúrán az F4 funkció billentyű lenyomásával is elérhetünk.

A válasz egy kis ablakban jelenik meg, és ha szerencsénk van, akkor az üzenet következő: „*No errors or warnings in model*”. Ha nincs szerencsénk, akkor a megjelenő hibaüzenet leírja a hibát. Hibaüzenet esetén használjuk a **Find** opciót a hiba megkereséséhez.

A modell szintaktikai ellenőrzése (**Run>Check Modell**) után, a szimuláció helyes működése és kényelmes követhetősége érdekében még a futás előtt célszerű beállítani a futás paramétereit. A **Run->Setup** menüpontokkal megnyitjuk a dialógusablakot, és a *Project Parameters* fülre kattintva a megnyíló ablakba beírhatjuk a projekt címét, és az analízist végző személy nevét.

A szimuláció futása alatt megállíthatjuk a szimulációt a **Run** eszközsorban a **Pause** gombra kattintva, **Run > Pause**, vagy az **Esc** billentyűvel. Ez ideiglenesen felfüggeszti a szimulációt, és a *User interrupted* üzenet jelenik meg a képernyőn az állapotjelző sorban. Ha a **Pause** módban duplán kattintunk az animációban látható entításokra, akkor megnyílik

az *Entity Summary* párbeszédablak, amely listázza az entitás attribútumok minden értékét. Ez nagyon hasznos lehet egy modell tesztelése közben. Ezenkívül a **Run** eszközsorban használhatjuk a **Step** gombot a szimuláció lépésenkénti végrehajtására. A **Go** gombbal pedig tovább futtathatjuk a megszakított szimulációt.

6.6.2. A modell ellenőrzése

Az ismertetett szimulációs modell helyes működésének ellenőrzése céljából, először a szántóföldi mérések adatait (6.1. táblázat) helyettesítettük a modellbe.

6.1. táblázat

A cukorrépa betakarítás és szállítás szimulációs modelljének input adatai

| Erőforrások | | |
|----------------|-------------------------------|---------------------------------------|
| Erőforrás neve | Művelet | Műveleti idő eloszlása és paraméterei |
| Holmer | Kiszedés 14 t | TRIA(6, 7, 10) min |
| Holmer | Kiszedés 17 t | TRIA(7, 8.5, 11) min |
| Holmer | Ürités pótkocsira | TRIA(1, 1.15, 2)) min |
| Holmer | Ürités prizmára | TRIA(1.2, 1.36, 2.4)) min |
| Ropa | Felszedés, tisztítás, rakodás | TRIA(4, 5, 7)) min |
| Volvo rakodó | Kanál telítés | TRIA(3, 9, 20)) sec |
| Volvo rakodó | Rakottmenet | TRIA(5, 14, 26)) sec |
| Volvo rakodó | Kanál ürités | TRIA(4, 8, 30)) sec |
| Volvo rakodó | Üresmenet | TRIA(4, 10, 27)) sec |

| Szállító eszközök | | |
|---------------------|-------------------------|---------------------|
| Szállítóeszköz neve | Sebesség | Üritési idő |
| Traktoros pótkocsi | TRIA(120,130,150) m/min | TRIA(2.5, 3, 5) min |
| Tehergépkocsi | TRIA(36,44.5,55.5) km/h | TRIA(2,3,4,6) min |

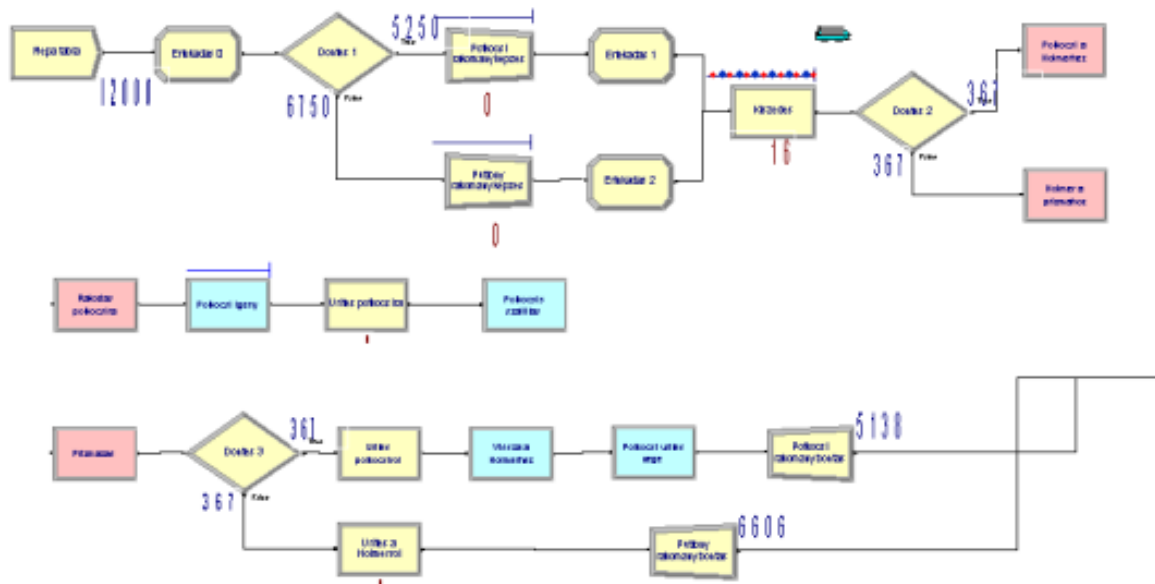
| Szállítási távolságok | |
|---------------------------|----------|
| Reláció | Távolság |
| Répaprizma - vasútállomás | 12 km |
| Holmer - répaprizma | 500 m |

6.6.3. Az eredmények megtekintése és értelmezése

A **Run>Go** menüopcióval, vagy az eszközsoron a **Go** gombbal elindítva a programot látható, hogy a rendszeren áthaladó a zöld, kék és sárga labdákon (a különböző méretű entitások) kívül, még az animált számlálók is megjelennek a szimuláció futtatása során. Egy számláló tartozik minden egyes *Create*, *Process* és *Dispose* modulhoz és kettő a *Decide* modulhoz. A *Create*, *Decide* és *Dispose* modulok számlálói eggyel növekednek valahányszor egy entitás belép a modulba. A *Process* modul esetében a számláló a pillanatnyilag a modulban tartózkodó entitások számát mutatja, beleértve az erőforrásra várakozó és már feldolgozás (megmunkálás) alatt álló entitásokat is. Ha a **Run > Fast-Forward** menü opciót (vagy a **Fast-Forward** gombot) választjuk, akkor ezeket a számlálókat (és az entitásokat a sorokban) csak a futás végén frissíti a program, akkor is, ha megszakítjuk a futást vagy nézetet váltunk a modellen. Az általunk végrehajtott szimuláció végén a betakarítási almodell képe a 6.5. ábrán látható.

A 6.5. ábráról leolvasható, hogy a 120 óra (7200 perc) alatt 12000 t répát generáltunk a *Create* modulban, és ebből a betakarítógép $367 \times 14 = 5138$ t és $367 \times 18 = 6606$ t, azaz

összesen 11744 tonnát szedett ki, és az a mennyiség a tábla szélein képzett prizmára került. Az adatokból könnyen kiszámítható a betakarítógép tömegteljesítménye $11744 \text{ t}/120 \text{ óra} = 97,87 \text{ t/óra}$, ami jól közelíti a valóságos értéket.

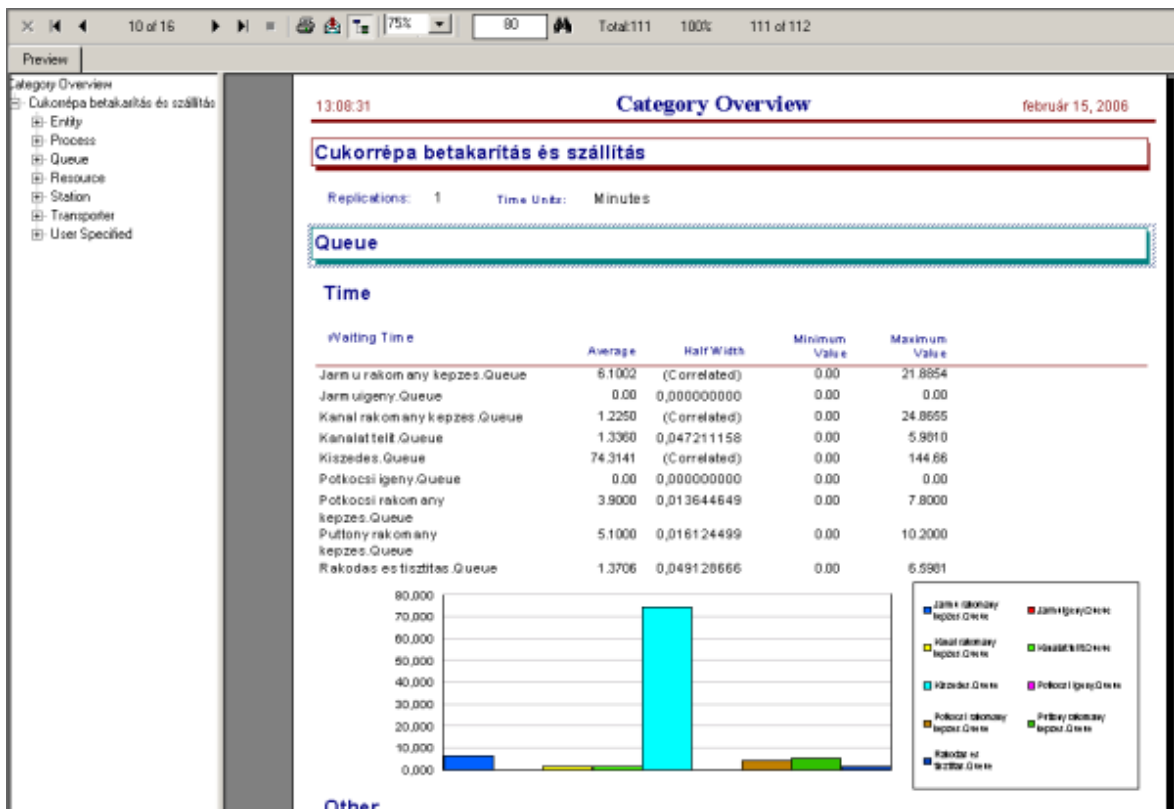


6.5. ábra. A betakarítási almodell szimuláció futásának végén

A modell futásának befejezésekor az **Arena** megkérdezi, hogy akarjuk-e látni az eredményeket. Ha a **Yes** gombra kattintunk, akkor megnyílik a **Category Overview Report** (Kategóriánkénti áttekintő jelentés) ablaka (6.6. ábra) (ez az alapbeállítás). Természetesen a jelentésben csak a **Run Setup Project Parameters** dialógusablakban kiválasztott statisztikák jelennek meg. Ha a dialógusablakban a **Statistics Collection** jelölő négyzetek valamelyikét üresen hagyjuk, akkor a hiányzó statisztika helyett „No Summary Statistics Are Available” (Az összesítő statisztika nem elérhető) üzenetet kapunk.

A jelentésablak bal oldalán, a **Preview** fül alatti fastruktúrájú jelentések között navigálhatunk vagy a gördítő gombokkal, valamint a jelentésablak bal felső sarkában lévő léptető gombokkal az egyik jelentésről a másikra ugorhatunk. A jelentés, mint már említettük, csak olyan statisztikát állít elő, amelyet a **Run>Setup** párbeszédablakban a **Project Parameters** fül alatt **Statistics Collection** területen bejelöltünk. A modellünk esetében ezek az *Entity*, *Process*, *Queue*, *Resource*, *Station*, *Transporter* és *User Specified* statisztikák. A *User Specified* (felhasználó által definiált) azért található, mert egy *Record* modul is elhelyeztünk a modellünkben.

A jelentésünkben háromféle statisztika található: ellenőrző (*tally*), folytonos (*persistent*) és számláló (*counter*). A negyedik statisztika típus a kimeneti adatok (*output*) csak a többszörös ismétlés (*multiple replication*) esetén érhető el. Az ellenőrző statisztika a *Record* modulok által gyűjtött folyamatidőt, várakozási időt és intervallum időket tartalmaz. A folytonos idejű statisztikákban a sorokban várakozó entitások száma, az eszköz használat és az eszköz kihasználtság szerepelnek. A számlálók a kumulált időt, belépő, kilépő és az összes feldolgozott entitások számát tartalmazzák. A felsorolt kategorizált statisztikák a 6.2. táblázatban tanulmányozhatók.



6.6. ábra. A Category Overview jelentésablak

Az ellenőrző és a folytonos statisztikákban egyaránt megjelenik a becslt átlag 95%-os konfidenciaszintjéhez tartozó konfidencia határ. A statisztikákban a *Half Width* a konfidencia intervallum felét jelenti, feltételezve, hogy a konfidencia határok a becslt átlaghoz viszonyítva \pm irányban szimmetrikusak. A statisztikák ezenkívül rögzítik a megfigyelt értékek minimális és maximális értékét is

<http://www.gek.szie.hu/tanszekek/geti/logisztika/lapok/Cjelentes.htm>.

Minden egyes ismétlés végén a program minden megfigyelt statisztika stacionárius (hosszú futás) várhatóértékére megkísérli kiszámítani a 95%-os konfidenciaszinthez tartozó konfidencia határokat, és ehhez az ún. „batch mean” eljárást használja. Az **Arena** először ellenőrzi, hogy elegendő adat gyűlt-e össze a kritikus statisztikai hipotézis (korrelálatlan tételek) vizsgálatához, ami a „batch-means” módszerhez szükséges. Ha nincs elég adat, akkor „Insufficient” (nem elégséges) felirat jelenik meg a jelentésben, és a jelentés nem tartalmazza a konfidencia intervallumot. Ha már van elegendő adat a korrelálatlan tételek teszteléséhez, de a vizsgálat hibát jelez, akkor „Correlated” (korrelált) felirat jelenik meg, és nincs konfidencia határ. Végül, ha elegendő adat gyűlt össze, és a hipotézisvizsgálat is sikeres, akkor a konfidencia határok és a várhatóérték is megjelenik a jelentésben. Az **Arena** így kizárja a megbízhatatlan konfidencia határok használatát.

Hibás eredményekre vezet, ha a rövid futási idő alapján próbálunk konklúziókat levonni, ezért nagyon fontos a futás hosszának és az ismétlések számának helyes meghatározása.

6.6.4. Az eredmények értékelése

A szimulációval elérhető elemzéseket és eredményeket nagyon röviden, csak a lehetőségeket felvillantva mutatjuk be, és ehhez a program által készített és a részletes zárójelentésben található <http://www.gek.szie.hu/tanszekek/geti/logisztika/lapok/Cjelentes.htm> 6.2 táblázatát használjuk fel.

A táblázat első része az entitásokról az **Entity** alcím alatt közöl információkat, amelyből megtudhatjuk, hogy egy entitás (Repa 1 t) a rendszeren 129,23 perc alatt halad keresztül. Ez az idő a következők szerint oszlik meg: a műveleti idő (VA Time, *Value Added Time*) 13,65 perc, a szállítási idő (Transfer Time) 27,35 perc és a várakozási idő (Wait Time) 88,22 perc. Megjegyezzük, az utóbbi tartalmazza a „Repa tabla” nevű *Create* modul és a „Kiszedés” nevű *Process* modul közötti sorokban keletkező várakozásokat is, amelyek csak a modell működéséből fakadó várakozások, ténylegesen nem jelentkezők.

Az **Entity Other** alcím alatti számok a belépő, a kilépő és a még feldolgozás alatt álló (WIP *Work In Process*) entitásokról tájékoztatnak. (A belépő és a kilépő mennyiség különbsége egyenlő a WIP mennyiséggel.)

A **Process** cím alatt a folyamatok egy entításra eső műveleti, a szállítási, várakozási és összes ideje látható. Az adatokat vizsgálva megállapíthatjuk, hogy a Holmer betakarítógép műveleti ideje sokkal nagyobb, mint a ROPA tisztító rakodógépé (8,27 perc, illetve 5,31 perc), aminek a ROPA rosszabb kihasználtsága (lásd később) köszönhető.

Az **Accumulated Time** cím alatt az előzőleg felsorolt, egy entításra eső idők kumulált értékei találhatóak. Emlékeztetőül, a szimulációs idő 7200 perc volt, amely alatt a betakarítógép 6069 percet, a tisztító rakodógép 2827 percet, a VOLVO rakodó pedig 1726 percet működött. A közvetlen ürítés összes ideje 610, a pótkocsiról való ürítés ideje 508 perc.

A jelentés következő szakaszában a **Queue** cím alatt a sorokban átlagosan eltöltött időről és a sorokban átlagosan várakozó entítások számáról, vagyis a sorok átlagos hosszáról tájékozódhatunk.

A *Kiszedes.Queue*, *Potkocsi rakomany kepes.Queue* és a *Puttony rakomany kepes.Queue* sorokat a korábban említett okok miatt nem kell vizsgálni. E sorokban várakozó entítások (répák) ugyanis még nincsenek kiszedve.

A *Jarmu rakomany kepes.Queue*, a *Jarmuigeny.Queue*, a *Kanal rakomany kepes.Queue*, a *Kanalat telit.Queue* a *Potkocsi igeny.Queue* és a *Rakodas es tisztitas.Queue* sorokban az átlagos várakozási idő és a sorok hossza nagyon kicsi, 0...1,37 perc, illetve 0...1,99 entitás.

Az erőforrások működésére, kihasználtságára vonatkozó adatokat a **Resource** cím alatt találunk. A jelentés szerint a betakarítógép kihasználtsága közel 100%-os, ezzel szemben a ROPA 39,28 %, a VOLVO rakodó pedig 23,97 % kihasználtsággal rendelkezik. A betakarítógép összesen 735 db 14, illetve 18 tonna rakománynak megfelelő mennyiségű répát szedett ki, a ROPA 533 db pótkocsis szerelvényt rakott meg, a VOLVO rakodó pedig 1943 db rakodási ciklust végzett a szimuláció ideje alatt.

Végül a jelentésben említést érdemelnek még a **Transporter** cím alatti kihasználási mutatók. A **Number Busy** részben kiderül, hogy a 6 járműből átlagosan csak 1,49 működik, ami miatt a kihasználtságuk mindössze 24,77 %. Ez azt jelenti, hogy modellben megadott 12 km-es távolság esetén a járművek száma a jobb kihasználás érdekében csökkenthető. A szántóföldön működő traktoros pótkocsi kihasználtsága: 63 %, jónak mondható.

A jelentés alapján a betakarítási rendszerről a következőket állapíthatjuk meg. A ROPA tisztító rakodógép és a VOLVO rakodó akár két betakarítógépet is képes kiszolgálni. Az illeszkedési problémák miatt az alrendszereket (betakarítás, tisztítás-rakodás-szállítás, vagonrakodás) ajánlatos egymástól függetlenül működtetni. Erre az alrendszerek között pufferolási céllal létesített prizmák lehetőséget adnak. A pufferek méretét a rendelkezésre álló hely, a tárolási idő, stb. befolyásolhatja.

6.2. táblázat

| Távolság [km] | Gépjármű szám | Felszedő-tisztító rakodógép kihasználási tényezője | Gépjárművek kihasználási tényezője | Rakományok száma [db/10 műszak] |
|---------------|---------------|--|------------------------------------|---------------------------------|
| 6 | 2 | 0,4011 | 0,9488 | 545 |
| | 4 | 0,4011 | 0,4750 | 545 |
| | 6 | 0,4011 | 0,3167 | 545 |
| 8 | 2 | 0,3456 | 0,9973 | 471 |
| | 4 | 0,3991 | 0,5744 | 545 |
| | 6 | 0,3991 | 0,3829 | 545 |
| 12 | 2 | 0,2548 | 0,99973 | 349 |
| | 4 | 0,3996 | 0,7750 | 545 |
| | 6 | 0,9408 | 0,5166 | 545 |

6.3. táblázat

| Távolság [km] | Gépjármű szám | Felszedő-tisztító rakodógép kihasználási tényezője | Gépjárművek kihasználási tényezője | Rakományok száma [db/10 műszak] |
|---------------|---------------|--|------------------------------------|---------------------------------|
| 6 | 2 | 0,4196 | 0,9991 | 574 |
| | 4 | 0,8248 | 0,9985 | 1114 |
| | 6 | 0,9994 | 0,9979 | 1354 |
| 8 | 2 | 0,3457 | 0,9991 | 471 |
| | 4 | 0,6840 | 0,9985 | 926 |
| | 6 | 0,9744 | 0,9979 | 1312 |
| 12 | 2 | 0,2552 | 0,9991 | 349 |
| | 4 | 0,5071 | 0,9995 | 691 |
| | 6 | 0,7546 | 0,9979 | 1021 |
| | 8 | 0,9622 | 0,9973 | 1305 |

A tisztító rakodógép áttelepítésekor a szükséges járműszámot a szállítási távolság, illetve az elérhető sebesség függvényében célszerű minden alkalommal újra kalkulálni. Ehhez elegendő a **Szállítás** almodellt futtatni.

A kalkuláció nagyon egyszerűen elvégezhető. A kiszolgáló járművek számát és a szállítási távolságot változtatva figyeljük a felszedő-tisztító rakodógép és a járművek költség- és kihasználási mutatóinak a változását.

Például a betakarítógép teljesítőképességével megegyező entitás generálás (0,6 min/1 t répa), 6, 8 és 12 km szállítási távolság és változatlan sebesség esetén a különböző járműszámokhoz tartozó kihasználás mutatók a 6.2. táblázat szerint változnak.

Ha az entitás generálással a tisztító rakodógép működését nem korlátozzuk, pl. az érkezési időközt 0,2 min/1 t répa-ra csökkentjük, a többi paramétert pedig a 6.2. táblázattal megegyezően változtatjuk, akkor a 6.3. táblázatba foglalt eredményeket kapjuk.

A két táblázat adatait vizsgálva a járműszám növelése általában a tisztító rakodó kihasználását javítja és a járművek kihasználását rontja. Ez a tendencia csak akkor nem érvényesül, ha a tisztító rakodógép működését a géphez érkező entitások érkezési intenzitása korlátozza. Ezt tapasztalhatjuk például a 6.2. táblázat első három sorában, ahol 2, 4 és 6 jármű esetén a tisztító rakodógép kihasználási tényezője egyforma értékű: 0,4011, és a járművek kihasználási tényezője pedig a járműszám növekedésekor csökken.

A 6.3. táblázatban, ahol az érkezési időközt 0,2 min/1 t répara csökkentettük, vagyis a répa a tisztító rakodógép teljesítőképességénél nagyobb intenzitással generálódik, az említett tendencia minden sorban érvényesül. Ebben a szituációban a tisztító rakodógép működését

inkább a járművek száma korlátozza, vagyis a tisztító rakodógép időnként azért nem működik, mert járműre várakozik. Ezt a megállapítást a 6.3. táblázat utolsó négy sora támasztja alá, ahol jól látható, hogy 2, 4 és 6 jármű esetében a tisztító rakodógép kihasználtsága relatíve alacsony, 8 jármű esetén viszont nagyon jó, miközben a járművek kihasználtsága sem romlik számottevően.

A 6.3. táblázat utolsó két sorát összevetve a részletes zárójelentés 6.2 táblázatának <http://www.gek.szie.hu/tanszekek/geti/logisztika/lapok/Cjelentes.htm> Resource alcíme alatti adatokkal, bizonyítottan tekinthetjük azt a hipotézisünket is, hogy a betakarítási folyamat fázisait célszerű függetleníteni egymástól. A jelentésben 12 km szállítási távolság és 6 jármű esetén a továbbított rakományok száma 533 volt, ezzel szemben a 6.3. táblázat szerint ugyanolyan feltételek mellett 1021 rakomány továbbítható, azaz a ROPA tisztító rakodógép teljesítőképessége megduplázható.

Az eredmények könnyen beláthatóak, és a költségek ismeretében hasonlóan vizsgálhatók a kihasználási mutatókkal arányos költségmutatók is.

7. CUKORRÉPA ÁTVEVŐ DEPÓK SZÁMÁNAK ÉS HELYÉNEK MEGHATÁROZÁSA

7.1. Bevezetés

A cukorrépa a szántóföldtől a cukorgyárig többféle módon juttatható el. Ismeretes a közvetlen (egytagú) átrakás nélküli és az összetett (többtagú) szállítás. Az utóbbi esetben a feladat végrehajtásában többféle szállító jármű és rendszerint több közlekedési alágazat (mezőgazdasági vontató, tehergépkocsi, vasút) működik együtt. A szállítás mellett a cukorrépa felvásárlás második legmunkaigényesebb művelete a tisztítás. A tisztítás során leválasztott szennyező anyagok elszállítása és kezelése, különösen akkor, ha a tisztítást a cukorgyár területén végzik nem kis feladatot jelent. E probléma kiküszöbölhető a szántóföld közelébe, ún. depókba telepített mobil tisztítók alkalmazásával. Ebben a fejezetben arra keresünk és adunk választ, hogy a szállítás szempontjából optimális depószám, illetve a depók helye miként határozható meg.

7.2. A fokozott tisztítású rakodás és szállítás

A 4. és 5. fejezetekben ismertetett szállítási formának mindegyikének vannak előnyei és hátrányai, amelyeket a teljesség igénye nélkül az eljárások bemutatása során ismertettünk. Ezek közül kitüntetett figyelmet érdemel a szakmai zsargon által fokozott tisztítású rakodásnak nevezett eljárás, amely a korábban alkalmazott módszerek előnyeit egyesíti.

Ennek a szállításnak a lényege, hogy a tábláról a répát közúti vagy mezőgazdasági járművekkel szállítják a tároló depókba, ahol azt prizmázzák, majd nagy kapacitású (150-450 t/h), magajáró, ún. tisztító rakodó gépekkel tisztítják és rakják közúti járművekre. A depó és a gyár távolságától, illetve a szállítási lehetőségektől függően a depókból közvetlen közúton, vagy két lépcsőben, közút-vasút kombinációval kerül a répa a cukorgyárba. A gyárba szállított fokozott tisztítású répa a mennyiségi és minőségi átvétel után azonnal feldolgozható. A fokozott tisztítású rakodás előnyeit és depók kialakításának fontosabb szempontjait a részletes zárójelentés

<http://www.gek.szie.hu/tanszekek/geti/logisztika/lapok/Cjelentes.htm> 4.1.2. pontjában ismertettük.

7.3. Modell a depók optimális helyének kijelölésére

A depók optimális helyének meghatározása egy ún. többkörzetes és kétlépcsős telephely kutatási probléma, amelyben az ismeretlen koordinátájú depók és az ismert koordinátájú cukorgyár(ak) a centrumoknak, a cukorrépa táblák pedig a kiszolgált telephelyeknek felelnek meg. A szállítás első lépcsője a cukorrépatáblák és a depók, a második lépcsője pedig a depók és cukorgyárak közötti mozgatás.

A megoldandó feladatban ismertek a kiszolgált telephelyek (táblák) kapacitásai és táblák középpontjának koordinátái. A kapacitás a tábla területéből és a termésátlagból számítható. A depók száma megegyezhet a tisztító rakodógépek számával, vagy esetleg más szempontokat mérlegelve magunk is előírhatjuk. A rakodógépek ugyanis áttelepíthetők, azonban szállítási megfontolások alapján célszerűbbnek látszik az 1 rakodógép-1 depó elv alkalmazása. A tisztító-rakodógép áttelepítése ugyanis jelentős időkieséssel jár. A modellben feltételezzük az 1 rakodógép-1 depó elv alkalmazását [7].

A depó kapacitását a rakodógép teljesítőképessége (180-220 t/h) határozza meg. A tapasztalat szerint egy rakodógép egy szezonban biztonsággal 100 ezer tonna répát képes megtisztítani, ami 60 t/ha-os termésátlag esetén 1700 ha termőterület kiszolgálását jelenti. A cukorgyárak olyan, a szokásostól eltérő tulajdonságú centrumok, amelyeknek nemcsak a kapacitásai, hanem koordinátái is ismertek.

A megválaszolendő kérdések, hova helyezük a depókat, hogyan körzetesítsünk, azaz az egyes cukorrépa táblákról mely depókba szállítsunk.

Az elmondottak alapján a feladat megoldására alkalmas **matematikai modell** feltételrendszere és célfüggvénye a következő:

$$(1) \quad X_{ij} \geq 0, \quad Y_{ik} \geq 0, \quad \text{ahol } i=1,2,\dots,n, \quad j=1,2,\dots,m, \quad k=1,2,\dots,l$$

$$(2) \quad \sum_i X_{ij} = t_j$$

$$(3) \quad \sum_j X_{ij} \leq f_i$$

$$(4) \quad \sum_i Y_{ik} \leq r_k$$

$$(5) \quad \sum_j t_j \leq \sum_i f_i$$

$$(6) \quad \sum_j t_j \leq \sum_k r_k$$

$$(7) \quad Q = \sum_i \sum_j X_{ij} \sqrt{(x_j - u_i)^2 + (y_j - v_i)^2} + \sum_i \sum_k Y_{ik} \sqrt{(\xi_k - u_i)^2 + (\eta_k - v_i)^2} \rightarrow \min,$$

ahol:

n a depók száma,

m a cukorrépa táblák száma,

l a cukorgyárak száma,

X_{ij} a j -edik cukorrépa tábláról az i -edik depóba szállított mennyiség,

Y_{ki} az i -edik depóból a k -edik cukorgyárba szállított mennyiség,

t_j a j -edik cukorrépa tábla kapacitása (egyenlő a cukorrépa tábla termésátlagával),

f_i az i -edik depó kapacitása,

r_k a k -edik gyár kapacitása,

f'_i az i -edik depóba szállított mennyiség,

$\mathbf{r}_i(u_i, v_i)$ az i -edik depó koordinátái,

$\mathbf{r}_j(x_j, y_j)$ a j -edik cukorrépa tábla koordinátái,

$\mathbf{r}_k(\xi_k, \eta_k)$ a k -edik cukorgyár koordinátái,

Q a szállítási munka.

A feladat megoldása gyakorlatilag a cukorrépa táblák körzetekbe sorolását, és az egyes depók koordinátáinak (u_i, v_i) megkeresését jelenti, vagyis a feladatot egykörzetes centrum problémákra vezetjük vissza.

7.1. táblázat

A kombinációs tábla felépítése

| | T_1 | T_j | T_m | D_1 | D_i | D_n | | Korlát |
|-------------|----------|----------|----------|----------|----------|----------|--------|--------|
| D_1 | c_{11} | c_{1j} | c_{1m} | M | M | M | \leq | f_1 |
| D_i | c_{i1} | c_{ij} | c_{im} | M | M | M | \leq | f_i |
| D_n | c_{n1} | c_{nj} | c_{nm} | M | M | M | \leq | f_n |
| G_1 | M | M | M | d_{11} | d_{1i} | d_{1n} | \leq | r_1 |
| G_k | M | M | M | d_{k1} | d_{ki} | d_{kn} | \leq | r_k |
| G_l | M | M | M | d_{l1} | d_{li} | d_{ln} | \leq | r_l |
| f_j | t_1 | t_j | t_m | f'_1 | f'_i | f'_n | | |
| Differencia | k_1 | k_j | k_m | k_1 | k_i | k_n | | |

A megoldás első lépése a körzetesítés, amely a cukorrépa táblák depóhoz való rendelését jelenti. Ezt a legegyszerűbben az ún. kombinációs tábla segítségével végezhetjük el (7.1. táblázat), amelyben a depók (D_i) és a cukorrépa táblák (T_j) közötti c_{ij} távolságok, a gyárak (G_k) és a depók (D_i) közötti d_{ij} távolságok, valamint a cukorrépa táblák (t_j), a depók (f_i) és a gyárak (r_k) kapacitásai szerepelnek.

A c_{ij} mátrix elemei a

$$c_{ij} = \sqrt{(x_j - u_i)^2 + (y_j - v_i)^2},$$

a d_{ki} mátrix elemei pedig a

$$d_{ki} = \sqrt{(\xi_k - u_i)^2 + (\eta_k - v_i)^2}$$

összefüggéssel számíthatók. Ehhez természetesen az első iteráció előtt önkényesen felvesszük az $\mathbf{r}_i(u_i, v_i)$ depó koordináták kezdő értékeit.

A körzetekbe soroláshoz a Vogel-Korda-féle módszerhez hasonlóan oszloponként differenciákat képezünk (k_i), ami az oszlop két legkisebb elemének a különbsége.

A körzetesítést először a D_i-T_j partícióban végezzük el, vagyis először a depókhoz rendeljük a cukorrépa táblákat. A programozást abban az oszlopban kezdjük, ahol a differencia a legnagyobb. Az oszlop legkisebb eleméhez a lehető legnagyobb mennyiséget (X_{ij}) rendeljük, majd a programozást a következő legnagyobb differenciájú oszlopban folytatjuk. Azt az oszlopot, illetve sort, ahol elfogyott a készlet, elhagyjuk. Ha sort hagyunk el, akkor újra számítjuk a differenciákat. A sorokban ügyelünk arra, hogy a megadott felső korlátokat (f_i) ne lépjük túl.

A depók és a cukorrépa táblák egymáshoz rendelése után a

$$\sum_j X_{ij} = f'_i, \quad i=1,2,\dots,n$$

összefüggéssel kiszámítjuk a depókba ténylegesen szállított mennyiségeket, ezeket beírjuk a kombinációs táblába. Az előző partícióban követett módszer szerint elvégezzük a körzetesítést a G_k-D_i partícióban is, azaz meghatározzuk Y_{ki} , az i -edik depóból a k -adik cukorgyárba szállított mennyiségeket, miközben a cukorgyárakhoz (G_k) rendeljük a depókat (D_i). A körzetesítés eredményét a 7.2. táblázat mutatja, amelyben teljesülnek a $\sum_j X_{ij} \leq f_i$, $\sum_i X_{ij} = t_j$, $\sum_i Y_{ki} \leq r_k$ és a $\sum_k Y_{ki} = f'_i$ feltételek.

7.2. táblázat

A szállított mennyiségek mátrixa

| | T_1 | T_j | T_m | D_1 | D_i | D_n | | Korlát |
|-------|----------|----------|----------|----------|----------|----------|--------|--------|
| D_1 | X_{11} | X_{1j} | X_{1m} | M | M | M | \leq | f_1 |
| D_i | X_{i1} | X_{ij} | X_{im} | M | M | M | \leq | f_i |
| D_n | X_{n1} | X_{nj} | X_{nm} | M | M | M | \leq | f_n |
| G_1 | M | M | M | Y_{11} | Y_{1i} | Y_{1n} | \leq | r_1 |
| G_k | M | M | M | Y_{k1} | Y_{ki} | Y_{kn} | \leq | r_k |
| G_l | M | M | M | Y_{l1} | Y_{li} | Y_{ln} | \leq | r_l |
| f_j | t_1 | t_j | t_m | f'_1 | f'_i | f'_n | | |

A körzetekbe sorolás után az egyes körzetek úgy vizsgálhatók, mint különálló, egykörzetes feladatok, és a koordináták-menti centrumnyomozás iterációs képleteivel a centrumok $\mathbf{r}_i(u_i, v_i)$ helyei meghatározhatók. A koordináták-menti centrumnyomozás iterációs képletei az i -edik körzetben:

$$u_i^{(k+1)} = \frac{\sum_j X_{ij} x_j / c_{ij}^{(k)}}{\sum_j X_{ij} / c_{ij}^{(k)}},$$

$$v_i^{(k+1)} = \frac{\sum_j X_{ij} y_j / c_{ij}^{(k)}}{\sum_j X_{ij} / c_{ij}^{(k)}},$$

ahol a $c_{ij}^{(k)} = \sqrt{(x_j - u_i^{(k)})^2 + (y_j - v_i^{(k)})^2}$, és k az iterációk száma.

Minden iterációs lépés után megvizsgáljuk a

$$\left| u_i^{(k+1)} - u_i^{(k)} \right| \leq \varepsilon,$$

$$\left| v_i^{(k+1)} - v_i^{(k)} \right| \leq \varepsilon$$

feltételeket, ahol az ε tetszőlegesen kicsi szám. Ha a feltételek teljesülnek, akkor befejezzük, különben folytatjuk az iterációs eljárást.

Az új centrumkoordinátákkal ($\mathbf{r}_i(u_i, v_i)$) új kombinációs táblát készítünk, és újra körzetekbe soroljuk a cukorrépa táblákat. Az eljárás akkor fejeződik be, ha két egymást követő kombinációs tábla azonos körzeteket eredményez. Meg kell jegyezni, hogy az eljárás sikere nagymértékben függ a kezdőértékek felvételétől. Előfordulhat ugyanis, hogy csak lokális minimumhelyet kapunk. Ez ellen a legegyszerűbben úgy védekezhetünk, hogy a

számításokat különböző indulóadatokkal megismételjük. Megnyugtató eredményt csak akkor remélhetünk, ha eltérő centrum sorozatok ugyanazt a megoldást szolgáltatják.

A végső megoldásból megtudhatjuk a depók (D_i) optimális helyét jelölő koordinátákat ($r_i(u_i, v_i)$), a depók tényleges kapacitásait (f_i'), az i -edik depóhoz tartozó cukorrépa táblákat (T_i), továbbá a k -adik cukorgyárhoz tartozó depókat.

7.4. Modell alkalmazása

A bemutatott modell alkalmazásának előfeltétele a modellben definiált objektumok, a cukorrépa táblák és az átvevőhelyek (cukorgyárak, vasútállomások, stb.) koordinátáinak ismerte. Ezek a koordináták az inputjai annak a számítógép programnak, amelyet Microsoft Access programozási környezetben kódoltunk. A számítógép program a modell adatbázisát kezelő űrlapokból, lekérdezésekből, és a számításokat végző modulokból áll.

A bemenő adatok meghatározásához először feltérképeztük a cukorrépa táblákat és az átvevőhelyeket. Erre a célra elsődleges adatgyűjtési eljárást, mesterséges holdakon alapuló helymeghatározást használtunk. A GPS rendszerrel 3 dimenziós helymeghatározást, időmérést és sebességmérést végezhetünk földön, vízen és levegőben. A rendszer pontossága jellemzően méteres nagyságrendű, de differenciális mérési módszerekkel (DGPS) mm-es pontosságot lehet elérni, akár valós időben is. A műholdas helymeghatározás gyakorlatilag időmérésre visszavezetett távolság meghatározás. A rádióhullám kibocsátási és beérkezési idejének pontos mérésével, továbbá a rádióhullámok terjedési sebességének ismeretében kiszámítható a jelforrás és a vevő közötti távolság.

7.4.1. A GPS használata és az adatok gyűjtése

A GPS eszközökről és a helymeghatározásról

A földi referencia rendszerben, hivatalosabban a geoid felületén, földrajzi helyzetünket a szélesség-hosszúság alapú koordináta rendszer segítségével fejezzük ki. Az egyenlítővel párhuzamosan futó, a sarkok felé egyre rövidülő köröket szélességi köröknek vagy földrajzi szélességnek nevezzük. A mindkét sarkponton áthaladó, egyenlő hosszúságú és az egyenlítővel derékszöget bezáró köröket hosszúsági köröknek vagy földrajzi hosszúságnak nevezzük. A Föld nem szimmetrikus természete miatt, a koordináta-rendszer felépítéséhez szükséges paraméterek közötti különbségekből adódóan több úgynevezett geodéziai dátum jött létre, amelyek kisebb-nagyobb eltérésekkel a Föld felületéhez próbálják igazítani saját pontosságukat. A térképészet fejlődése során az egyes országok és régiók létrehozták a saját környezetüknek leginkább megfelelő dátumot, így manapság több mint 100 különböző viszonyítási rendszer létezik. A globális helymeghatározás igénye és a folyamatosan fejlődő technikai háttér megkövetelte egy, az egész Föld területén használható geodéziai dátum létrehozását. Az 1984-ben nemzetközi megállapodások során létrejött globális koordináta rendszert WGS84-nek (World Geodetic System) nevezik, és ez a rendszer képezi alapját a ma legfejlettebb, széles körben elterjedt helymeghatározó rendszereknek. (Megjegyezzük: technikai oldalról szemlélve a GPS által visszaadott pozíciónak nincs köze semmilyen földi referencia rendszerben értelmezett koordinátához.) A továbbiakban WGS koordinátaként a GPS által visszaadott, az egész Föld felületén értelmezhető földrajzi koordinátákra utalunk.

A GPS rendszerrel történő adatgyűjtés céljára a kereskedelem ma már számos eszközt és megoldást kínál. Alapvetően az adatgyűjtő eszközök egy antennával ellátott GPS vevőkészülékből, és egy ehhez vezetékkel, vagy vezeték nélkül (infravörös, bluetooth) kapcsolódó mobil számítógépből állnak. A számítógép lehet PDA (Personal Digital Assistant) vagy notebook. A vevőkészülék és a számítógép gyakran egy egységet képez. A GPS

A modell megoldására szolgáló algoritmust az adatbázissal azonos környezetben az MS Access-ben programoztuk.

A modell megoldásának eredményei depók $r_i(u_i, v_i)$ koordinátái A számított koordináták természetesen csak tájékoztató adatoknak tekinthetők, mivel a számítások során számos egyéb szempontot, korlátozó tényezőt, például az útviszonyokat nem vesszük figyelembe. A depók végleges helyeit a cukorgyár szakemberei a számított koordináták és az említett egyéb szempontok mérlegelése alapján jelölték ki.

8. ÖSSZEFOGLALÁS

A kutatás elméleti megalapozását a téma irodalmának feldolgozásával kezdtük. E munka egyrészt a cukorrépa és a cukor történetének feldolgozására, másrészt a közelmúlt eseményeinek az elemzésére és a jelenlegi helyzet feltárására irányult. Az irodalom feldolgozás keretében részletesen foglalkoztunk a hazai cukorrépa termelés változásaival (fajták, termőterület, gépesítettség, termelési eredmények, stb.), illetve a termeltetés (szerződések, a termeléshez szükséges anyagok biztosítása, stb.) és a felvásárlás (a betakarítás és a beszállítás programozása, minősítés, stb.) problémáival. Tanulmányoztuk a hazai és a külföldi szakirodalomban és gyakorlatban fellelhető cukorrépa betakarítási és beszállítási módszereket.

Áttekintettük a cukorrépa termelést és cukorgyártást szabályozó uniós rendtartás folyamatban lévő változásait, és azok várható következményeit. Az EU elődje, az EGK cukorrendtartása 1968 óta kiszámítható, magas jövedelmet biztosított a cukorrépa termelőknek és a cukorgyáraknak. Ez a szabályozás a magas belső árakra, a termelést szabályozó kvótákra és a magas import vámokra és a belső feleslegek támogatott exportjára épült. A kedvező helyzetet a magyar gazdák és cukorgyártók nem sokáig élvezhették, mert az Európai Bizottság álláspontja szerint a cukorreform, ami 2006-ban már el is kezdődött elkerülhetetlen, mert a kvótán felül termelt C cukor és az import miatt az unió évi 4-5 millió tonna cukorfelesleggel rendelkezik.

A cukorrépa termeltetés és átvétel hazai helyzetének felmérése, illetve az alkalmazott módszereknek a külföldi fejlesztési tendenciákkal való összevetése érdekében tanulmányoztuk a Szerencsi Cukorgyár Rt jelenlegi termeltetési és felvásárlási gyakorlatát. A hazai cukorgyárak a megfelelő mennyiségű és minőségű alapanyag biztosítása érdekében korszerű technikai háttérrel támogatott termeltetési felügyelői hálózatot működtetnek. A termeltetés folyamatának nyomon követésére, valamint a cukorrépa átvételének és gyártelepre való beszállításának megszervezésére az alkalmazott integrált vállalatirányítási rendszerhez kapcsolódó információs és nyilvántartási programokat használnak. Általánossá vált gyakorlat, hogy a cukorrépa táblák pontos helyzetét és méretét, az átvétel helyétől, illetve a gyártól való távolságát műholdas térinformatikai rendszer (*GPS*) segítségével határozzák meg. A mért adatok alapján a termelői körzetről digitális térképet készítenek, amely vizuálisan is szemlélteti a táblák elhelyezkedését és méretét, a kijelölt répaátvételi helyeket (depókat), valamint a lehetséges szállítási útvonalakat.

A legkorszerűbb kommunikációs eszközöket és a számítástechnikát alkalmazó rendszerek közül kiemelten foglalkoztunk a *MIR* (Modulare Integrierte Rübenlogistik) elnevezésű logisztikai rendszerrel, és a hazai bevezetés lehetőségével. A *MIR* négy alapfunkciója: a répatáblák, répatároló prizmák koordinátáinak a meghatározása; optimális beszállítási terv készítése számítógépen; online adatforgalom a közreműködők között; a répakiszedés, a rakodás és a beszállítás számítógépes irányítása.

Rendszereztek a cukorrépa szállítás lehetséges megoldásait. Ismeretes a közvetlen (egytagú) átrakás nélküli és az összetett (többtagú) szállítás. Az utóbbi esetben a feladat végrehajtásában többféle szállító jármű és rendszerint több közlekedési alágazat (mezőgazdasági vontató, tehergépkocsi, vasút) működik együtt. Részletesen elemeztük, az ún. fokozott tisztítású rakodás és szállítás bevezetésének előnyeit, illetve a bevezetés során megoldandó feladatokat. A cukorrépa kiszedést követő műveleteket, amelyek ugyan a fejlesztés egyik szűk területét jelentik, a szállítás és a szállítási költségek csökkentése szempontjából vizsgáltuk. Megállapítottuk, hogy az eljárás bevezetésének egyik neuralgikus pontja a cukorrépadepók helyének, számának és méretének a megválasztása. Ezért feladatunknak tekintettük a depók optimális helyének meghatározását, ami egy ún. többkörzetes és kétlépcsős telephely kutatási probléma, és amelyben az ismeretlen koordinátájú depók és az ismert koordinátájú cukorgyár(ak) a centrumoknak, a cukorrépa táblák pedig a kiszolgált telephelyeknek felelnek meg. A szállítás első lépcsője a cukorrépatáblák és a depók, a második lépcsője pedig a depók és cukorgyárak közötti mozgatás. A megválaszolható kérdések, hova helyezük a depókat, hogyan körzetesítsünk, azaz az egyes cukorrépa táblákról mely depókba szállítsunk. A megoldásához kifejlesztettünk a probléma matematikai modelljét, majd elkészítettük annak adatbázisát és számítógépes programját is.

Formulákat dolgoztunk ki a betakarítógép fogás hosszának és a szükséges kiszolgáló járművek számának determinisztikus számítására a különböző működési stratégiákra.

A kutatásban kiemelten foglalkoztunk a cukorrépa betakarítás, prizmázás, tisztítás, rakodás és szállítás modellezésével. A szántóföldi mérések eredményeiből megállapítható, hogy ezek a műveletek nagy számú és különböző eloszlású valószínűségi változóval jellemezhető sztochasztikus folyamatokat alkotnak, amelyek a klasszikus értelemben vett matematikai modellekkel már nem írhatók le. Az ilyen folyamatok modellezésének egyetlen járható útja a szimuláció.

A tanulmányban bemutattuk az **Arena** környezetben fejlesztett, az üzemi megfigyeléseken alapuló, diszkrét szimulációs modell felépítését és a modellalkotás lépéseit. A különböző működési szituációkban a lehetséges konfigurációkkal futtatott modellből nyert eredményeket a legígéretesebb megoldások kiválasztása céljából elemeztük.

Megállapítottuk azt is, hogy a Holmer betakarítógép, a ROPA felszedő-tisztító rakodógép és a VOLVO rakodó jelentős teljesítmény különbsége miatt célszerű a teljes folyamatot fázisokra bontani és azokat függetleníteni egymástól. Bemutattuk, hogy az adott kapacitású vezérgépekhez (a betakarítógép és a felszedő-tisztító rakodógép) a kihasználási tényező függvényében miként lehet a kiszolgáló eszközök (traktoros pótkocsi, járművek, rakodógép) szükséges kapacitását (darabszámát) meghatározni.

A részletesen ismertett eljárással:

modellezhetjük a cukorrépa betakarítási folyamatokat annak érdekében, hogy definiáljuk, dokumentáljuk és kommunikáljuk azokat;

szimulálhatjuk a rendszerünk működését, azért, hogy megértsük a komplex kapcsolatokat és azonosítsuk a fejlesztési lehetőségeinket;

a dinamikus animációval **láthatóvá tehetjük** a rendszerünk működését;

számtalan lehetséges alternatív konfiguráció esetén **elemezhetjük** a rendszerünk működését, és ezek közül biztonságosan kiválaszthatjuk a legjobbbat.

Végezetül köszönetet mondunk a kutatást támogató partnereinknek, a Szerencsi Cukorgyár Rt-nek és az adonyi Március 21. Mezőgazdasági Szövetkezetnek, illetve e cégek azon dolgozóinak, akik munkánkat segítették.

A Szerencsi Cukorgyár Rt-vel kötött kutatási együttműködési szerződés keretében az Rt rendelkezésünkre bocsátotta matematikai modellezéshez szükséges információkat, és hozzájárult ahhoz, hogy a kutatócsoport tagjai a beszállítóknál, a közvetítőpontokon és az átvevőhelyeken megfigyeléseket végezzenek, illetve adatokat gyűjtsenek.

Kiemelt köszönettel tartozunk az adonyi szövetkezet munkatársainak, akik nemcsak lehetővé tették, hanem segítették is a szántóföldi mérések végzését.

Gödöllő, 2006. február 27.

IRODALOM

1. Arena Standard Edition User's Guide. Rockwell Softwer, 157. p.
2. Arena Professional Edition Reference Guide. Rockwell Softwer, 298. p.
3. Arena Variables Guide. Rockwell Softwer, 70. p.
4. **Benkő J.:** Logisztikai tervezés. (Mezőgazdasági alkalmazásokkal) Dinasztia Kiadó, Budapest, 2000. 199 p.
5. **Benkő J.-Soós P.-Szüle Zs.- Balogh A.:** Cukorrépa átvevő depók számának és helyének meghatározása. MTA Agrár Műszaki Bizottsága Kutatási Tanácskozása, Gödöllő, 2004. január 20-21.
6. **Benkő J.-Soós P.:** Új módszerek a cukorrépa logisztikában. Mezőgazdasági Technika, ILV. évf. 10. sz. 2004. 2-5 p.
7. **Benkő J.–Soós P.– Szüle Zs.– Balogh A.:** Determination of the number and the places of sugar-beet receiving repositories. Hungarian Agricultural Engineering, Nr.17/2004. 88-90 p.
8. **Benkő J. - Soós P.- Szüle Zs.- Balogh A.:** Cukorrépa betakarítás logisztikai fejlesztésének lehetőségei. MTA Agrár Műszaki Bizottsága Kutatási Tanácskozása, Gödöllő, 2005. január 18-19.
9. **Benkő J. - Soós P.- Szüle Zs.- Balogh A.:** Die Möglichkeit der Entwicklung bei Zuckerrüben-Logistik. MicroCAD Nemzetközi Tudományos Konferencia, Miskolc 2005. március 9-10.
10. **Benkő J.-Soós P.:** Új módszerek a cukorrépa logisztikában. Logisztikai évkönyv 2005 (Szerk.: Knoll I.), Magyar Közlekedési Kiadó, Budapest, 2005.
11. **Benkő J.–Kiss L.–Kondás J.:** Cukorrépa depók telepítési kérdései. Cukoripar, LVIII. évfolyam, 1. 2005. január-március, 12-16 p.
12. **Benkő J. - Soós P - Balogh A. - Miklós A. – Rácz P.:** A cukorrépa betakarítás és szállítás vizsgálata diszkrét szimulációval. MTA Agrár Műszaki Bizottsága Jubileumi XXX. Kutatási és Fejlesztési Tanácskozása, Gödöllő, 2006. január 24.
13. **Detrekői Á.- Szabó Gy.:** Bevezetés a térinformatikába. Nemzeti Tankönyvkiadó, Budapest, 1995. 250 p.
14. **Ewers, A.-Schieche, B.:** SAGA – Satellitendaten und GIS in der Rübenlogistik. Zuckerrüben, 49. (6) 2000.
15. **Fischer B.:** A cukorvertikum helyzete Magyarországon. Cukorrépa XXI. évf., 1 szám 2003.
16. **Fischer B.:** Ágazatunk helyzete, kihívásai. Cukoripar, LVI évfolyam, 2 szám, 2003. április-június, 54-55 p.

17. **Fórián Z.:** Térségünk és az Eu cukoripara. Cukoripar, LVI. évfolyam 2. szám, 2003.
18. **Füzy J.:** Gépesítés fejlesztési irányok és hatékonyság a cukorrépa betakarításban. Cukoripar, LII évfolyam, 2 szám, 1999. április-június, 76-80 p.
19. **Füzy J.:** Az egymenetes cukorrépa – betakarítás szállításszervezése. Cukorrépa, XX. évf., 2. szám, 2002.
20. **Füzy J.:** A cukorrépa betakarítás gépesítése. Agrár unió 6. szám, 2003. augusztus-szeptember.
21. **Gerber, P.:** Anfuhrlogistik der Diamant-Zucker. Zuckerrübe 6/2001, 356-357 p.
22. **Jung, G.-Kammerling, B.:** A Nordzucker AG kooperál a LIZ-zel (Nordzucker kooperiert mit LIZ) Zuckerrübe 2/2002 (51.jg.)
23. **Kelton, W., D.-Sadowski, R., P.-Sturrock, D.,T.:** Simulation with Arena. Mc Graw Hill Higher Education, International Edition, 2004. 668 p.
24. **Kemmer, H -Grothaus H, P:** Trends in der Rübenlogistik. Zuckerrübe, 6/2000. (49.Jg.)
25. **Koczka Z.:** Világpiaci kihívások és az EU cukorrendtartása. Cukoripar, LV. évfolyam, 2. szám, 2002.
26. **Koczka Z.:** A magyar cukoripar az EU csatlakozás küszöbén Cukoripar LVI. évfolyam, 1. szám, 2003.
27. **Koczka Z.:** „Az Európai Únió cukoripari rendtartása és a világpiac helyzete a keleti bővítés küszöbén.” Cukoripar, LVI. évf., 2. szám, 2003. 57-60 p.
28. **Koczka Z.:** Az EU cukoriparának változó arca. Cukoripar, LVI. évfolyam 3. szám, 2003.
29. **Kondás J.:** Logisztikai folyamatok a Szerencsi Cukorgyár Rt. mezőgazdasági igazgatóságánál. (Szakmérnöki diplomaterv), GATE, GTK. Vállalatgazdálkodási Szakmérnöki Szak, Gödöllő, 1998.
30. MapBasic Development Enviroment Reference Guide Version 7.0. MapInfo Corporation, 2002. 684 p.
31. MapInfo Professional v7.5 User Guide (Abridged). MapInfo Corporation, 2003. 496 p.
32. OptQuest for Arena User's Guide. Rockwell Softwer, 54. p.
33. **Pommerehne, C.:** Was kosten Rübenvorreinigung und Rübentransport? Zuckerrübe, 4/1999. (48. jg.)
34. **Pommerehne, C.:** Mietenpositionierung per GPS. Zuckerrübe, 3/2002 (51.jg.)
35. **Pommerehne, C.–Kemmer, H.:** Nordzucker AG MIR-System Nordzucker. „Modulare Integrierte Rübenlogistik-innovative Datenvernetzung beim Zuckerrübentransport” Zuckerrübe 2/2003 (52.jg.)
36. **Posch K. (szerk):** „A 2002. évi cukorrépatermesztés főbb mutatói” Beta-Kutató és fejlesztő kft. éves kiadványa.
37. **Prezenszki J.:** Logisztika (Bevezető fejezetek). BME, Mérnöktovábbképző Intézet, Budapest, 1995.
38. **Pritsker, A. Alan B.:** Introduction to Simulation and SLAM II. Halsted Press Book, John Wiley & Son and System Publishing Corporation, 1986.

39. Sajtóinformáció: Hosszútávra szóló szállítási jogokat kaptak a cukorrépa termesztők. Cukoripar, LV.évf., 2.szám, 2002.
40. **Schneider, T.:** Union Zucker und Nordzucker KG weihen gemeinsam Zuckerversandzentrum und neue Flüssigzuckeranlage ein. Zuckerrüben, 49. (6) 2000.
41. **Shmilliár M:** A cukorrépa termesztése. Budapest, Akadémiai Kiadó, 1965.
42. **Soós P. -Benkő J -Szüle Zs.- Balogh A.:** Új módszerek, eljárások a cukorrépa felvásárlásban és átvételben. MTA Agrár Műszaki Bizottsága Kutatási Tanácskozása, Gödöllő, 2004. január 20-21.
43. Szerencsi Cukorgyár Rt. Minőségügyi Folyamatleírásai, Szerencs, 2003.
44. **Szilágyi L.:** A fokozott tisztítású rakodás gyakorlata a Kabai Cukorgyár Rt.-nél. Cukorrépa termesztési és termeltetési tanfolyam. Cukoripari Egyesülés, Budapest 2001. 104-107 p.
45. **Szüle Zs.-Füzy J.:** A cukorrépa betakarító gépek nemzetközi bemutatója Selingenstadt-ban. Cukorrépa, 1997/2, 11-15 p.
46. Taylor II. Simulation User's Guide. F&H Simulation Inc.
47. **Zsugyelik G.:** Az Európai Unió közös cukorpiaci szabályozása Cukorrépa, XXI. évf. 2003/1.
48. **Soós P.:** Cukorrépa termesztés korszerűen. Budapest, Mg. Kiadó 1976. 306 p.
49. <http://www.nol.hu/cikk> Kész, de fájdalmas az EU cukorreformja
50. <http://reggel.hu/index> A jövőben harmadával is olcsóbb lehet a cukor
51. <http://www.stop.hu> Az uniós cukorreform hátrányos, de nem feltétlenül jár munkahelyek megszűnésével
52. <http://www.origo.hu/uzletinegyed/hirek/hazaihirek> Árthat a cukorreform Magyarországnak.
53. <http://www.origo.hu/uzletinegyed/hirek/hazaihirek>. Csökken a cukorrépa termelői ára.
54. <http://www.fvm.hu/> Tájékoztató a cukoripari rendtartás reformjának vitájáról.
55. <http://www.fvm.hu/> Cukorreform.