Alkalmazott Matematikai Lapok **31** (2014), 1-40.

ÜTEMEZÉSI FELADATOK AZ AUTÓBUSZOS KÖZÖSSÉGI KÖZLEKEDÉS OPERATÍV TERVEZÉSÉBEN: EGY ÁTTEKINTÉS

ÁRGILÁN VIKTOR, BALOGH JÁNOS, BÉKÉSI JÓZSEF, DÁVID BALÁZS, GALAMBOS GÁBOR, KRÉSZ MIKLÓS, TÓTH ATTILA

A közlekedési társaságok számára lényeges szempont költségeik racionalizálása, amit legkönnyebben az operatív költségeik csökkentésével valósíthatnak meg. Ez a járatok, a járművek optimalizált ütemezésével is elősegíthető. Egy ilyen optimalizálási feladat nagyon komplex, ezért a műveletek ütemezését három fázisra bontva tárgyaljuk. Ezek a járműütemezés, a vezetőütemezés és a műszakkiosztás feladatai. Ezek a részfeladatok általában NP-nehéz problémák, ezért az elmúlt évtizedig közepes méretű feladatok optimális megoldása sem volt lehetséges. A számítási sebesség, az alkalmazott modellek és az azokat megoldó algoritmusok olyan jelentősen fejlődtek, hogy lehetővé vált nagyobb méretű feladatok kezelése is. A cikkben áttekintjük az említett részfeladatok megoldására szolgáló legfontosabb módszereket. Tárgyaljuk azok különböző matematikai modelljeit, hatékony megoldási lehetőségeikkel együtt. A speciális korlátozó feltételek szerepét saját tapasztalatainkon keresztül vizsgáljuk. Utalunk a busz- és vezetőütemezési probléma általunk tanulmányozott és bevezetett megoldási módszereire is.

1. Bevezetés

A közösségi közlekedési szolgáltató vállalatok kiadásainak nagy részét az operatív költségek alkotják. Ez jórészt a járműflotta költségéből, a járművek üzemanyag- és karbantartási költségéből, valamint a járművezetők fizetéséből tevődik össze. Ebből következően az operatív költségekben mutatkozó megtakarítás számottevő javítást adhat költségvetésükben. A leggyakrabban használt módszer ezeknek a költségeknek a csökkentésére egy hatékony, számítógéppel támogatott információs rendszer kialakítása és használata. Az IKT (információs és telekommunikációs technológia) fejlődésének köszönhetően napjainkra szinte minden közösségi közlekedési társaság – modern vállalatirányítási környezetet biztosítva – rendelkezik saját információs rendszerrel.

Ezeknek a rendszereknek a fő funkciói az üzleti alkalmazásokon túl (úgymint könyvelés, számvitel stb.) olyan modulokat is tartalmaznak, amelyek

– előkészítik a járművek ütemezését a vállalat által kiszolgált vonalakhoz,

– illesztik a járművek és a járművezetők műszakjainak ütemezését a vonalakhoz,

– képesek a járműflotta nyomon követésére és monitorozására egy-egy nap során,

– jelzik a diszpécsernek a szokatlan eseményeket (meghibásodás, késés stb.),

– nyomon követik a járműflotta járműveinek állapotát,

és más hasonló tevékenységeket.

Az IKT-környezet alapvető háttérként szolgál a hatékony logisztikai irányításra (lásd pl. [56]), ugyanakkor a folyamatban jelen van egy másik elvárás, amely azt kívánja, hogy találjuk meg a rendszer azon részeit, amelyek az operatív, műveleti költségek szempontjából csökkenthetők.

Jónéhány ipari döntéstámogatási eszköz létezik, amely a logisztikai rendszertervezési és optimalizálási feladatok komplett megoldását célozza. A gyakorlatban ennek ellenére kiderül, hogy sok vállalatspecifikus részlet, korlátozó feltétel van jelen, amelyeket az általános rendszerek nem kezelhetnek egységesen, de amelyek fontosak a közlekedési vállalatok számára. Példaként említjük, hogy amennyiben a közlekedési társaság alternatív üzemanyagú járműveket is alkalmaz flottájánál, akkor ezek ütemezésénél figyelembe kell venni a szaknyelven rádiusznak nevezett, egy tankolással megtehető kilométerek számát, amely jóval kevesebb lehet, mint a hagyományos üzemanyagokkal működő járművek futási teljesítménye. Ilyen eseteket vizsgáltak [62]-ben és [53]-ban. Alternatív üzemanyagú járműveknél fontos tényező a tankolás időtartama is. Amíg hagyományos üzemanyagoknál (pl. dízelolaj) a tankolás kb. 5 perc, addig *CNG (compressed natural gas)* vagy más üzemanyagok esetén a feltöltési idő ennek többszöröse is lehet. Ilyen jellegű feladatok kezelését ismertettük néhány cikkünkben (lásd pl. [9]). Azért, hogy megfelelő járműütemezés készüljön, ezeket a feltételeket számításba kell venni a napi ütemezést elkészítő szoftvernél. Legjobb tudomásunk szerint napjaink ütemező szoftverei nem kezelnek ilyen jellegű feltételeket, vagy azokat csak korlátozott mértékben veszik figyelembe.

Követve a közösségi közlekedési szolgáltatás fejlesztésének általános módszereit, egy fejlesztés fő horizontjai a következők:

– *stratégiai tervezés*, ennek fő eleme a buszok, járatok útvonalának meghatározása (buszvonalak kialakítása),

– *taktikai tervezés*, melynek legfontosabb eredménye a menetrend elkészítése és

– *operatív tervezés*, a szolgáltatás ellátásához a buszok és a vezetők, a műszakok ütemezése.

Az irodalomban létezik másféle felfogás is, a feladatokat lehet másképpen csoportosítani. Borndörfer [14] a stratégiai és a taktikai tervezést közös, ún. *„szolgáltatástervezési”* fázisba vonja össze. Megemlít olyan kapcsolódó – a szolgáltatástervezés körébe sorolható – feladatokat (pl. jegyárazás, tarifatervezés és -kialakítás), amelyek az utazási igényeket is meghatározhatják. Az ilyen jellegű feladatokat – és a hozzájuk tartozó modelleket – ebben a cikkben nem tárgyaljuk.

Annak, hogy mi csak az említett operatív tervezési részfeladatok tárgyalására szorítkozunk, a terjedelmi korlátokon túl az is az oka, hogy a közösségi közlekedési szolgáltató vállalatoknak a többi említett stratégiai és taktikai tervezési feladatra kevés befolyásuk van: ezek általában az állami kormányzati vagy helyi önkormányzati (pl. városi) szervek által meghatározottak. Azon kívül, hogy a járatok milyen sűrűn kövessék egymást, más előírások is vonatkozhatnak a vonalakra, és azon belül bizonyos járatokra is. Például a vonalak kapacitására, vagy a szolgáltatást ellátó járművek speciális tulajdonságaira is lehetnek előírások. Egy jellemző példa az, hogy melyik vonalon milyen időközzel közlekedjenek alacsonypadlós járművek.

Ezen túlmenően viszont szinte teljesen a közlekedési társaságok döntésétől függ, hogy a saját járműflottájukat hogyan ütemezik, és a járművezetőiket hogyan osztják be műszakokba rövid- és hosszútávon egyaránt.

Az már ebből a bevezetőből is látszik, hogy a költségek csökkentése ebben a környezetben egy komplexen összefonódott problémacsokor, amelynek megoldása globális optimalizálási módszereket igényel. Mivel mindezekre egy teljesen összetett megközelítés megvalósíthatatlannak látszik, ezért olyan részproblémákra osztjuk a feladatot, amelyek eléggé izoláltak ahhoz, hogy ezeket a gyakorlatban kezelni tudjuk. Ezek

– a buszvonalak útvonaltervezése,
– a menetrendtervezés,
– az operatív ütemezés.

Megjegyezzük, hogy ennek ellenére a tervezési fázisok és az ütemezési fázis, ha nem is szorosan, de általában mégis hatnak egymásra annak érdekében, hogy globálisan hatékonyabb – vagy legalább lehetséges – megoldás legyen nyerhető.

A tudományos közösség évtizedekkel ezelőtt felismerte, hogy az operatív tervezési problémák optimalizált megoldása nagy kihívást jelentő, érdekfeszítő feladat. Ezeket a feladatokat tárgyalva sok eredmény született (lásd például a [12, 13, 17, 23, 27, 54] cikkeket), amelyek különböző modelleket és eltérő hatékonyságú algoritmusokat tárgyalva foglalkoztak a témával. Hamar kiderült, hogy a legtöbb részprobléma NP-nehéz, ami időben sokszor reménytelenné teszi az optimális – vagy közel optimális – megoldások megtalálását a gyakorlatban felmerülő problémák esetén. Már közepes méretű feladatok – például néhány százezer fős lakosságú városok – esetén is elég összetett, komplex feladattal állunk szemben, amely még összetettebbé válik, ha a részleteket is figyelembe kell vennünk. Még bonyolultabbá teszi a problémát, ha olyan korlátokat is kezelni kell, amelyeket a jogszabályok, a szakszervezetek, az egyéni igények és egyéb gyakorlati feltételek határoznak meg. Ezek például a járművezetők vezetési idejére, munkaidejére, pihenőidejére és munkaközi szüneteire vonatkozó előírások. Ilyen további korlátozó feltételeket adhatnak a telephelyek kötöttségein, kapacitásán és egyéb feltételein keresztül más gyakorlati részletek, egészen odáig, hogy a vezetőknél megfelelő mértékben legyenek vegyítve a kedvelt és nem kedvelt műszakok. Ehhez az utóbbihoz kapcsolódóan megemlítjük, hogy például Abbink és szerzőtársai 2007-ben publikált cikke [1] tárgyal egy vasúti műszakkiosztási példát.

Nem célunk sem a hálózati útvonal-, sem a menetrend tervezés tárgyalása. Mindezekhez Desaulniers [24] áttekintő tanulmányát javasoljuk, amely jó bevezetés, kiindulópont lehet ilyen jellegű problémák megoldásának tanulmányozásához.

Annak ellenére, hogy felvázoltuk a (buszos) tömegközlekedési rendszerek általános struktúráját, mi a jelen tanulmányban az operatív tervezés egyes fázisait vizsgáljuk. Ezen belül a buszflotta járműveinek ütemezésére, a járművezetők műszakonkénti ütemezésére és beosztására fogunk koncentrálni. A munkaerő ütemezését további részfeladatokra bontjuk: külön fejezetben tárgyaljuk a járművezetők ütemezését és a műszakok kiosztását a járművezetők között.

A cikk felépítése a következő. Mint látni fogjuk, a legutóbbi időkig kidolgozott eljárások a két ütemezési problémát egymás után végrehajtva oldják meg. A járműütemezés eredményeire támaszkodva keresnek optimális – vagy közel optimális – megoldást a vezető-ütemezési feladatok megoldására. Ezért a két ütemezési problémát elválasztva tárgyaljuk a második és a harmadik fejezetben. Azonban azt is látni kell, hogy ma már egyre nagyobb teret nyernek azok a megközelítések, amelyek a jármű- és vezető-ütemezési problémát együtt kezelik, és egyszerre próbálják megoldani. Figyelembe véve az algoritmusokban bekövetkező javításokat és a hardverek rohamosan növekvő számítási kapacitását, nem kétséges, hogy ezek a módszerek is hamarosan beépülnek az újonnan kifejlesztendő interaktív rendszerekbe. Ezért külön fejezetet szentelünk a napjainkban kibontakozó hatékony integrált modellek kidolgozására szolgáló kutatások áttekintésének.

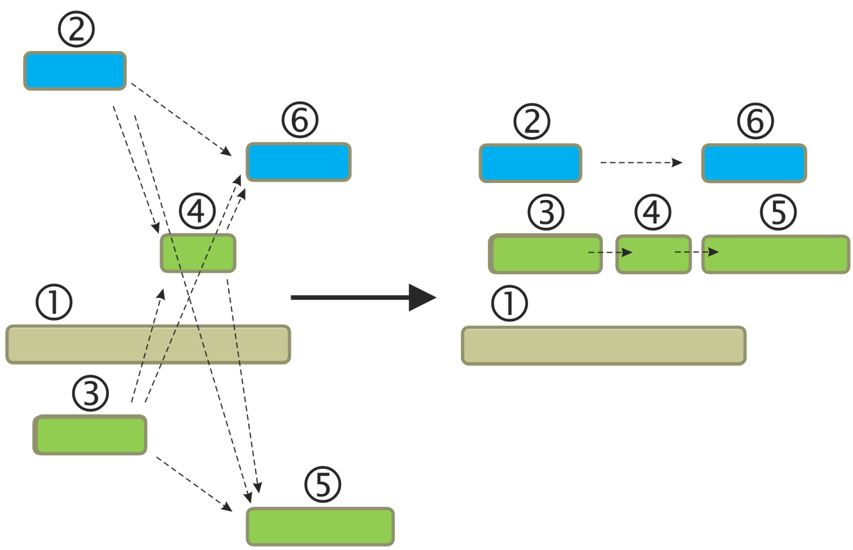
A következő fejezetekben először általánosan – moduláris szerkezetben – vizsgáljuk a feladatokat, majd alfejezetenként részletesen tárgyaljuk az ismert megoldási módszereket.

2. A járműütemezési feladat

2.1. A feladatról általában, definíciók, jelölések

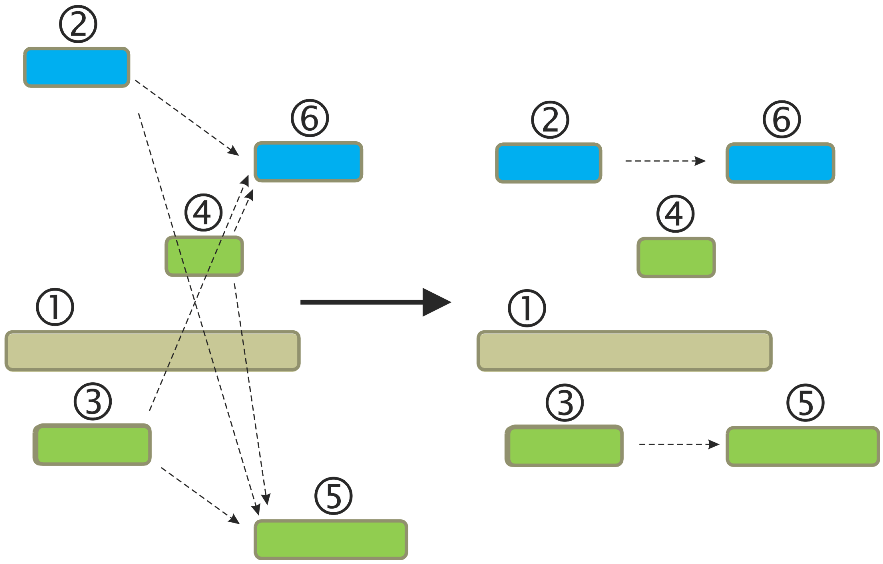
A járműütemezési problémánál *(Vehicle Scheduling Problem, VSP)* adott a *járműflotta járműveinek* és a menetrendi *járatoknak* a halmaza. Jelölje ezeket rendre két halmaz, $V = \{v_1, v_2, \ldots, v_{m^*}\}$ és $U = \{u_1, u_2, \ldots, u_{n^*}\}$. Egy V halmazba tartozó minden $v \in V$ jármű azonos típusba tartozik (pl. alacsony padlós és gázüzemű). Ha több – eltérő típusú – járművünk van, akkor a járművek halmazát diszjunkt V_i $(i = 1, \ldots, k)$ részhalmazokra bontjuk. Jelölje továbbá $c_{i,j}$ azt a költséget, amivel az i jármű a j járat által előírt tevékenységet elvégzi. A feladat az, hogy a járműflotta elemeit rendeljük hozzá az egyes járatokhoz úgy, hogy a hozzárendelés összköltsége minimális legyen. Az első pillanatban egyszerű hozzárendelési feladatnak tűnő probléma azért bonyolódik meg, mert nem kell minden járművet járathoz rendelni, és egy jármű – a járatok időbeli eltolódása miatt – több járathoz is hozzárendelhető. További korlátozó feltételt jelenthet az, hogy minden járatot pontosan egyszer kell végrehajtani, és – esetleg további feltételek alapján – minden jármű legyen képes végrehajtani azokat a járatokat, amelyekhez a megoldásunk során hozzárendeltük.

Minden $u_i \in U$ menetrendi járatra adott a járat $dt(u_i)$ *indulási* ideje és $at(u_i)$ *érkezési ideje*, valamint $dg(u_i)$ *indulási* és $ag(u_i)$ *érkezési földrajzi helye*. Az u_i és az u_j járatot *kompatibilisnek* (ebben a sorrendben egymás után végrehajthatónak, összefűzhetőnek) nevezzük, ha ugyanaz a jármű képes kiszolgálni egymás után őket, azaz $at(u_i) \le dt(u_j)$, és $dt(u_j) - at(u_i)$ kisebb, mint a $dg(u_j)$ és $ag(u_i)$ közti távolság megtételéhez szükséges idő. Az 1. és a 2. ábra egy-egy, néhány járatból álló sematikus szituációt mutat. A sematikus ábrákon szaggatott nyilak jelzik a kompatibilis járatpárokat (itt a vízszintes tengely reprezentálja az időt, a földrajzi helyek nincsenek ábrázolva), csak akkor kötünk össze szaggatott nyíllal két járatot, ha egymással kompatibilisek. Az 1. ábrán jelzett sematikus szituációban ilyen módon 3 jármű elég a 6 járat ellátásához, míg a 2. ábráén ehhez már 4 jármű kell.



1. ábra. 6 járatot ábrázoló sematikus feladat. A szaggatott nyilak jelzik az egymással kompatibilis járatokat. Az ábra jobb oldali része azt illusztrálja, hogy a feladat 3 járművel megoldható.

Az ütemezési időszak kezdetén a járművek *depókban* állnak, és az ütemezési időszak végén oda is térnek vissza. Depóba kerülhet egy jármű az ütemezési idő-



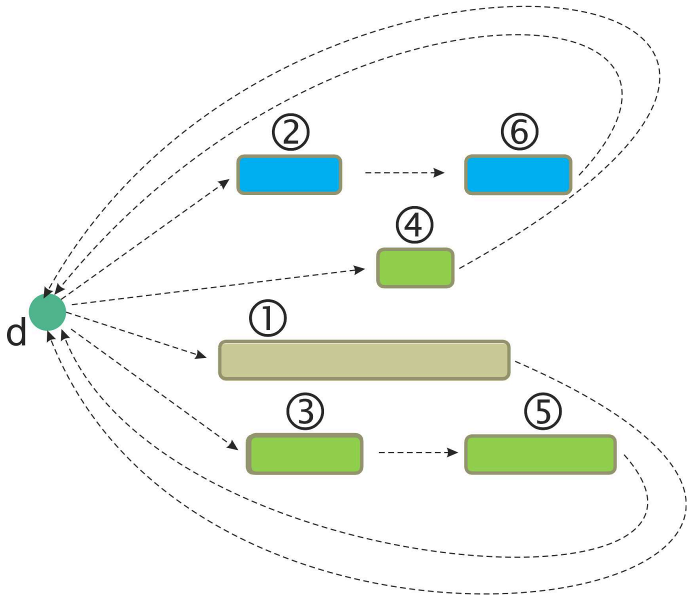
2. ábra. Az 1. ábráéhoz hasonló sematikus szituáció, de itt az ottaninál kevesebb járat kompatibilis egymással, így ez a feladat csak 4 járművel oldható meg.

szak bármely időpontjában, ha hosszabb ideig nem rendelünk hozzá járatot. Depó lehet garázs, parkolóhely vagy telephely, attól függően, hogy a jármű hol parkol.

A depók számától függően beszélhetünk egydepós (*Single Depot Vehicle Scheduling Problem*, SDVSP) és többdepós (*Multiple Depot Vehicle Scheduling Problem*, MDVSP) járműütemezési feladatról. A többdepós esetben az is elő lehet írva, hogy melyik járatot melyik depókhoz tartozó járművek hajthatják végre.

A menetrendi járatokon kívül, amelyek szállítanak utast, megkülönböztetünk olyan járatokat is, amelyek nem szállítanak utast. Ekkor beszélünk *rezsijáratról*. Ez utóbbiak közé tartozik a depóból történő ki- és beállás, valamint két kompatibilis menetrendi járat esetén az első járat érkezési és a második járat indulási földrajzi helye közötti átállás.

Egy jármű *ütemezése* járatoknak egy olyan lánca, amelyben minden egymást követő két menetrendi járat egymással kompatibilis. Általában egy jármű ütemezése a gyakorlatban egy jármű egy napi munkájának előírását jelenti, amit járműműszaknak (szakszóval járműfordának vagy eszközfordának) is nevezünk. A jármű egy érvényes ütemezése egy kiállási járattal kezdődik, és egy beállási járattal végződik. A 3. ábra a 2. ábrának megfelelő sematikus szituáció egy érvényes járműütemezésekké kibővített megoldását mutatja. Itt feltettük, hogy egy depó van (egydepós eset), és a depóból valamennyi járat indulási helyére és érkezési helyéről a depóba vezetnek depókiállási és -beállási járatok).



3. ábra. A 2. ábrán látható sematikus szituáció megoldása egy depó esetén a depó ki- és beállási járatokkal érvényes járműütemezéseket alkot.

A járműütemezés alapfeladata abból áll, hogy adjuk meg a járműflotta járműveinek ütemezését a fenti módon úgy, hogy minden egyes menetrendi járat hozzá legyen rendelve pontosan egy jármű ütemezéséhez, és minden menetrendi járat a megfelelő depók valamelyikéből legyen végrehajtva. (Természetesen lehetnek olyan járművek is, amelyek az adott, (pl. egy napi) ütemezésben nem vesznek részt.) Célfüggvényként kezelhetjük az ütemezésben használt járművek számának a minimalizálását, de definiálható más költségfüggvény is. A cél akkor a költségfüggvény minimalizálása. Ha a költségfüggvény a teljes ütemezés költségét reprezentálja, akkor tartalmaznia kell az egyes depókhoz tartozó átalányköltséget, valamint operatív költséget is. Az átalányköltségen azt a költséget értjük, amely abból keletkezik, hogy egy adott jármű a depóban rendelkezésre áll. Ez lehet a beszerzési, fenntartási, karbantartási stb. költségekből vetített átlag. Az operatív költség általában a megtett távolságokkal arányos, azonban különböző lehet attól függően, hogy melyik depóból származó jármű látja el az adott feladatot. A járatok operatív költsége attól is függhet, hogy rezsi-, vagy menetrendi járatról van-e szó, mivel különböző lehet a kilométerek „egységára”. Több modell képes úgynevezett depókapacitási korlátozó feltételek kezelésére is, azaz lehetséges megoldásnak csak olyan megoldásokat tekint, amely figyelembe veszi minden depó esetén az ahhoz tartozó járművek maximális számát.

A célfüggvényben az ütemezett járművek általános (pl. a fenntartási költségekből adódó) és utazási (menetrendi- és rezsijáratainak) költségei szerepelnek. Némileg könnyíti a feladatot, hogy mint említettük, a költség második komponense általában a megfelelő út hosszával arányos, ugyanakkor a menetrendi járatok és a rezsijáratok kilométereihez eltérő költségek is tartozhatnak. Természetesen a rezsijáratok költségei így függenek a megfelelő földrajzi helyek közötti távolságoktól, következésképpen eltérő lehet a rezsijáratok költsége attól függően, hogy egy adott menetrendi járat mely másik járatot követ.

Több alapvető matematikai modell létezik különböző (SDVSP, MDVSP) feladatok megoldására, amelyeket az előző évtizedekben dolgoztak ki. A következő alfejezetekben áttekintjük a feladatok néhány alapvető megoldási módszerét. A napjainkban talán legszélesebb körűen használt MDVSP-modellekben a probléma egy egészértékű többtermékes hálózati folyam problémaként fogalmazódik meg (lásd [13, 48, 54]). Ebben a modellben az optimális ütemezést egy egészértékű lineáris programozási feladat megoldásaként számíthatjuk ki. A probléma megfogalmazható halmazlefedési vagy halmazpartícionálási feladatként is (lásd például [43, 63]).

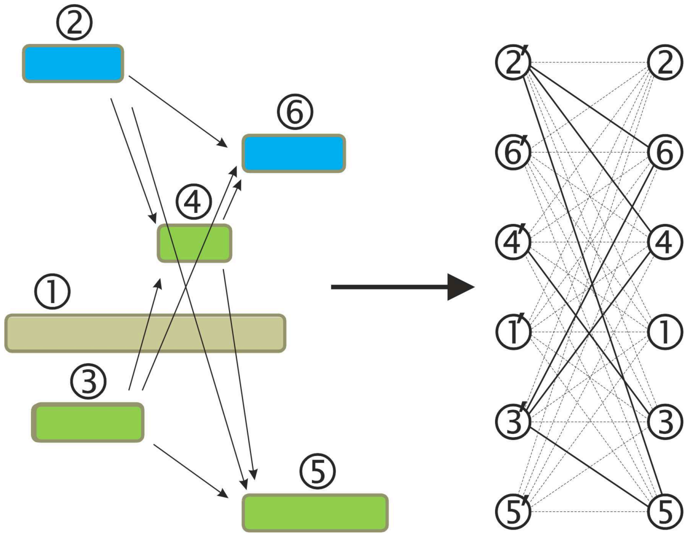
2.2. Az egydepós járműütemezési feladat

Az SDVSP-problémára adott első megoldási módszert Saha [64] publikálta. Az U járathalmaz elemeire egy részleges rendezést definiál, és bevezetésre kerül egy β rendezési reláció, mely szerint akkor szolgálható ki u_2 legálisan u_1 után, ha $ag(u_1) = dg(u_2)$, valamint $at(u_1) \leq dt(u_2)$. A megkötésekből látható, hogy ez a modell nem engedélyezi a különböző járatok között végrehajtott rezsimeneteket, így az itt meghatározott β reláció a fent definiált kompatibilitásnál gyengébb.

Az SDVSP-re több páros gráfon alapuló modellt is publikáltak. Ezekről bővebb áttekintés található Bunte és Kliewer [16] munkájában. Mi az alábbiakban Bertossi és társai [12] modelljét és megoldását ismertetjük, ahol szintén párosítási problémaként modellezik a feladatot. Vegyünk egy $G = (G_1, G_2, E)$ teljes páros gráfot, ahol G_1 és G_2 csúcshalmazok, minden $i \in G_1$ csúcs egy-egy járat érkezésének (érkezési földrajzi helyének), míg minden $j \in G_2$ csúcs egy-egy járat indulásának (indulási földrajzi helyének) felel meg. E jelöli az élhalmazt, valamint $|G_1| = |G_2| = |U| = n^*$, és $|E| = (n^*)^2$. Tegyük fel továbbá, hogy E két E_1 és E_2 részhalmazra osztható, amelyek:

$$E_1 = \{(i, j) \mid u_i \text{ és } u_j \text{ kompatibilis járatpár}\},$$
$$E_2 = E \setminus E_1.$$

A fenti gráf a következő módon értelmezhető (lásd a 4. ábrán egy egyszerű, sematikus feladat illusztrációját). Az $(i, j) \in E_1$ élek két kompatibilis u_i és u_j járat érkezési és indulási földrajzi helyei közti lehetséges rezsijáratot jelképezik,



4. ábra. Az 1. ábrán látható sematikus szituációhoz tartozó páros gráf Bertossi és szerzőtársai SDVSP-modelljében [12]. A folytonos, vastagabb vonallal rajzolt élek az E_1 élhalmazt, a szaggatott, vékonyabb vonallal rajzolt élek az E_2 élhalmazt ábrázolják itt.

míg minden $(i, j) \in E_2$ él két egymás utáni rezsijáratnak felel meg. Ezek közül az első az u_i járat érkezési földrajzi helyéről a depóba, majd a másik a depóból az u_j járat indulási földrajzi helyére. Ezek segítségével látható, hogy a G gráf egy M teljes párosítása egy lehetséges járműütemezést fog adni, melyben a járművek száma $|M \cap E_2|$.

Az (i, j) élekhez $c_{i,j}$ költségeket rendelve az SDVSP feladata megfeleltethető egy minimális költségű teljes párosítás keresésének a G gráfban.

- Ha $c_{i,j} = 1$ minden $(i, j) \in E_2$ esetén, és $c_{i,j} = 0$ egyébként, úgy a feladat megoldásával megkapjuk a járatok teljesítéséhez szükséges minimális eszközszámot.
- Ha $c_{i,j}$ értékei az adott rezsijáratok végrehajtásához szükséges költségek lesznek, akkor a feladat megoldása a minimális operatív költséget adja.

Természetesen az előbbi költségek valamely kombinációja is használható. A gyakorlati életben a probléma kiegészül még a járművek darabszámára adott korláttal. Legyen ez a korlát k. Ekkor egy korlátos párosítási feladatot kapunk, ahol a feladat a minimális költségű párosítást megkeresni a gráfban az $|M \cap E_2| \leq k$ feltétel mellett. A probléma formálisan a következőképpen adható meg: legyen x bináris változók vektora, ahol $x_{i,j} = 1$, ha az (i, j) él az M párosításhoz tartozik, különben $x_{i,j} = 0$, és legyen c a költségvektor. Az X lehetséges megoldások halmazát az alábbi feltételeknek megfelelő vektorok határozzák meg:

$$\sum_j x_{i,j} = 1, \quad i = 1, 2, \ldots, n^*,$$
$$\sum_i x_{i,j} = 1, \quad j = 1, 2, \ldots, n^*,$$
$$\sum_{(i,j)\in E_2} x_{i,j} \le k.$$

A célfüggvény

$$min_{(i,j)}\{cx \mid x \in X, \quad x_{i,j} \in \{0,1\}\}.$$

A fenti feladat minimális költségű hálózati folyamproblémaként is megoldható (lásd [16]). A hálózat konstruálása ekkor annyiban tér el a fent definiált gráftól, hogy további csúcsokat és éleket vezetünk be: a járatokat jelző csúcsokat a hálózatban kettébontjuk a járat indulását és a járat érkezését reprezentáló csúcsra, melyeket egy, a járatot jelző éllel kötünk össze. Ezeknek a járat éleknek az alsó és felső korlátja is 1 lesz, így biztosítva azt, hogy minden járat pontosan egy alkalommal kerüljön teljesítésre. Ennek az a következménye, hogy az ezeken az éleiken felmerülő költségek egy konstans többletként jelentkeznek, ami a feladat célfüggvényére nincs hatással. A járatok csúcsaihoz hasonlóan a depót jelképező csúcs helyett is egy depóindulási, illetve depóérkezési csúcsot vezetünk be. Ezeket egy 0 költségű éllel kötjük össze, mely a hálózat depókörfolyam éle lesz. Ha a fentiek szerint korlátozzuk a rendelkezésre álló eszközök számát, úgy ennek az élnek a kapacitása k lesz. Ezt a fajta formalizmust használva Ahuja és munkatársai bebizonyították, hogy a feladat erősen polinomiális időben megoldható [3].

2.3. A többdepós járműütemezési feladat

A járműütemezési feladat megoldására leggyakrabban használt modell az úgynevezett többdepós járműütemezési probléma (*Multiple Depot Vehicle Scheduling Problem*, MDVSP) modellje. A valós életben a különböző menetrendi járatokra és a járművekre speciális igények vonatkozhatnak. Az eltérő járműtípusok, valamint a járművek tartózkodási helye alapján a járműveket különböző depókba oszthatjuk, így a járatok kiszolgálása – az igényektől függően – különböző depókból történhet.

A többdepós járműütemezési problémát Bodin és szerzőtársai definiálták [13], majd Bertossi és szerzőtársai mutatták meg róla, hogy NP-nehéz feladat [12]. A modell értékét az adja, hogy tartalmazza a valós életbeli járműütemezési probléma legfontosabb komponenseit. A matematikai modellek ismertetése során jelöléseinkben a Löbel által használt terminológiát követjük [54].

A többdepós járműütemezési feladatnál minden menetrendi járat esetén a felhasználó megadja azokat a depókat, ahonnan az adott járat kiszolgálható. A gyakorlatban ez jelentheti például azt, hogy bizonyos járatok csak csuklós busszal láthatók el, és az is megadható, hogy mely esetben mely telephelyhez tartozzanak. Ezeket az előírásokat a telephely és az állomás elhelyezkedése, valamint a forgalom jellemzői határozhatják meg.

A következőkben az MDVSP különböző megoldási technikáit mutatjuk be. Ezek egy része többtermékes folyamproblémára vezeti vissza a feladatot, és egészértékű programozási (IP) feladatként oldja meg azt. A különbség az alapul szolgáló hálózat felépítésének módszerében rejlik. Ezek alapján megkülönböztetünk kapcsolatalapú és idő-tér hálózati modellt. Egy másik megközelítésben a problémát halmazpartícionálási, illetve halmazlefedési feladatként modellezik. Ezt a modellt elemezték például Ribeiro és Soumis [63], vagy Hadjar és szerzőtársai [43].

2.3.1. A kapcsolatalapú többtermékes hálózati modell

A kapcsolatalapú többtermékes hálózati folyam (*connection-based multicommodity network flow*) modell széleskörűen használt az MDVSP-probléma megoldására. Sok, a témába vágó kutatás fókuszált ennek alkalmazására. Intenzíven tanulmányozták a generált egészértékű programozási feladat megoldási módszereinek fejlesztésében rejlő lehetőségeket. Számos megközelítés alapul heurisztikus, közelítő módszerekre (lásd [23, 54, 57]), míg mások pontos megoldást szolgáltató, egzakt algoritmusokat tárgyaltak (lásd [49, 55]).

Mielőtt rátérünk a modell leírására, a korábban bevezetett jelölések mellé új definíciókat is meg kell adnunk. Jelölje D a depók halmazát, és $D_u \subseteq D$ egy u menetrendi járat depóhalmazát: ez azokat a depókat tartalmazza, amelyekből ki lehet kiszolgálni az u járatot. Jelölje $U_d \subseteq U$ azoknak a járatoknak a halmazát, amelyek kiszolgálhatók a d depóból. Hasonlóan, minden $d \in D$ esetén definiálunk két, $dt(d)$ és $at(d)$ – depóindulási és depóérkezési – csúcspontot a hálózatban; ezek szimbolizálják azt, hogy a jármű a d depóból indul, és oda érkezik vissza. A hálózat csúcspontjainak N halmazát ezek után a következő módon definiáljuk

$$N = \{dt(u) \mid u \in U\} \cup \{at(u) \mid u \in U\} \cup \{dt(d) \mid d \in D\} \cup \{at(d) \mid d \in D\}.$$

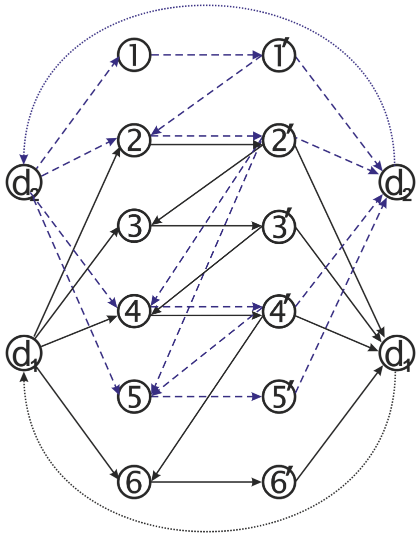
A hálózat éleinek definíciójához vezessük be a következő jelöléseket. Egy d depóhoz tartozó menetrendi járatok halmazához rendeljük a következő irányított éleket:

$$E_d = \{(dt(u), at(u)) | u \in U_d\}, \quad \forall d \in D.$$

Rendeljünk irányított élt a d depó minden u járatának érkezési idejét reprezentáló csúcsból az u-val kompatibilis – szintén d-beli – járatok indulási időpontjához rendelt csúcsba:

$$B_d = \{(at(u), dt(u')) \mid u, u' \in U_d \text{ kompatibilis járatok}\}, \quad \forall d \in D.$$

B_d élei rezsijáratok. További, a d depóhoz tartozó nem menetrendi járatok, amelyekhez éleket kell rendelni a hálózatban, alkotják a depóhoz tartozó kiállási



5. ábra. A kapcsolatalapú modell hálózata egy kétdepós feladat példája esetén. (A földrajzi helyek mellett ezen az ábrán az indulási és érkezési időpontokat sem reprezentáljuk, de csak azokat a csúcsokat kötjük össze éllel, ahol a megfelelő járatok kompatibilisek.)

és beállási élek halmazát:

$$R_d = \{(dt(d), dt(u)), (at(u), at(d)) \mid u \in U_d\}, \quad \forall d \in D.$$

Ha korlátot szeretnénk előírni az egyes depókban rendelkezésre álló járművek számára, akkor szükségünk van az ún. „*depókörfolyam*” élek halmazának definiálására is:

$$K_d = \{(at(d), dt(d))\}, \quad \forall d \in D.$$

Ezek alapján meg tudjuk adni a hálózat d depóhoz tartozó éleinek a halmazát:

$$A_d = E_d \cup B_d \cup R_d \cup K_d, \quad \forall d \in D,$$

és a gráf összes éleinek halmaza

$$E = \cup_{d \in D} A_d.$$

A kapcsolatalapú modell hálózatának felépítését az 5. ábra szemlélteti.

Ezen előkészületek után most már készen állunk arra, hogy definiáljuk az MDVSP-feladatot a $G = (N, E)$ hálózaton. Ehhez definiálunk egy egészértékű x vektort, amely egy többtermékes folyamként is tekinthető. A vektor dimenziója megegyezik a hálózat éleinek számával. A vektor $e \in E$ élhez tartozó komponensét x_e^d-vel jelöljük, ha az e él a d depóhoz tartozik ($e \in A_d$). Az x_e^d komponens értéke egy adott ütemezésben 1 lesz, ha az adott él benne van az ütemezésben, különben 0. Ez alól csak a depókörfolyam élek a kivételek, mert azok többször is szerepelhetnek egy ütemezésben.

Az x vektor segítségével olyan korlátozó feltételeket definiálunk, amelyek a feladat követelményeit biztosítják.

Ütemezésünkben biztosítani kell azt, hogy minden menetrendi járatot pontosan egyszer hajtsunk végre. Ez azt jelenti, hogy egy lehetséges megoldásban csak olyan járműütemezések szerepelhetnek, amelyeknek – a depókörfolyam élektől eltekintve – nincs közös élük. Ezt a következő feltételekkel érhetjük el:

$$\sum_{d \in D_u, e=(dt(u),at(u)) \in E_d} x_e^d = 1, \quad \forall u \in U.$$

Biztosítanunk kell azt is, hogy az ütemezési időszak végére minden jármű viszszaálljon egy depóba. Ez más megfogalmazásban azt jelenti, hogy ha egy adott depóhoz tartozó jármű egy – depótól eltérő valamelyik – csúcsra (állomásra) megérkezik, azt el is kell hagynia.

$$\sum_{e \in n^+} x_e^d - \sum_{e \in n^-} x_e^d = 0, \quad \forall u \in U, \forall n \in N,$$

ahol n^+ jelöli az $n \in N$ csúcsból induló élek halmazát, és hasonlóan, n^- az $n \in N$ csúcsba futó élek halmaza.

Ha vannak depókapacitást korlátozó feltételek, akkor azokat a depókörfolyam élekhez kell kapacitásként, vagyis a folyamértékre vonatkozó felső korlátként előírni. Ez azt jelenti, hogy ha k_d a d depóhoz tartozó azonos típusú buszok száma, akkor a feltételrendszerhez hozzá kell adni az

$$x_{at(d),dt(d)}^d \leq k_d$$

feltételt.

Bármely, a fenti feltételeket kielégítő folyam a feladat egy lehetséges megoldása lesz.

Amennyiben optimális megoldást szeretnénk kapni, definiálnunk kell egy célfüggvényt, amelynek a fenti feltételeket kielégítő optimális megoldását keressük. Ez az élekhez nemnegatív, valós értékű súlyokat rendelve történhet. Az él súlya az adott járat költségét reprezentálja. Amennyiben egyszerűen a járművek számát szeretnénk minimalizálni (ekkor a flottaminimalizálási feladatnak nevezett problémáról van szó), akkor a kiállási élekhez a többi él költségéhez viszonyítva nagyon nagy súlyokat kell rendelni. Ha c_e jelöli az e élhez rendelt költséget, akkor a feladat célfüggvénye:

$$\min \sum_{e \in E} c_e x_e.$$

Ez a fenti korlátozó feltételekkel együtt tekintve egy egészértékű (IP) programozási feladatot határoz meg, amelynek megoldása – bizonyos értelemben – rutinfeladat. Ennek megoldásával kapcsolatban a legfőbb felmerülő probléma az, hogy a hálózatnak túl sok éle lehet. Gondoljunk csak egy több ezer menetrendi járatot tartalmazó hálózatra. Ez már egy közepes, néhány százezer lakosú városnál is előfordulhat, és egy ilyen esetben a lehetséges rezsiátmenetek száma milliós nagyságrendű is lehet. Ennek az oka az, hogy a kapcsolatalapú hálózat minden élt tartalmaz, ahol akár csak elméletileg is lehetséges (rezsi)átmenet (az adott két járat elméletileg kompatibilis egymással). A végső megoldásba ezeknek ugyan csak egy csekély hányada kerül be, de nem lehetséges ezek elhagyása, mert azzal esetleg elveszíthetjük a feladat optimális megoldását is.

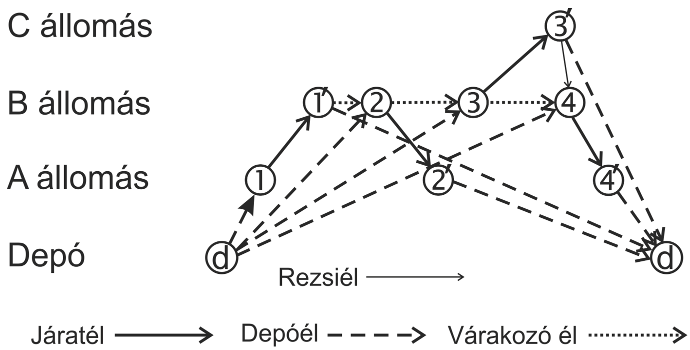
2.3.2. Az idő-tér hálózati modell

Az élek számának csökkentése egy új, módosított modellel történhet. Ezt tárgyaljuk a következőkben. Az *idő-tér hálózati (time-space network, TSN) modellt* Kliewer és szerzőtársai vezették be [48] közúti közösségi közlekedési feladatok kapcsán. Annak ellenére, hogy az idő-tér modellt légiközlekedési feladatoknál (repülőjáratok ütemezésénél) már használták korábban (lásd pl. [44]), a járműütemezés területén [48] az első idő-tér módszert tárgyaló publikáció.

A modell fő vívmánya, hogy nagyobb méretű, a gyakorlatban előforduló problémákat is meg lehet vele oldani.

A modell két dimenziót használ, ezek az idő és a tér. A tér szó jelentése itt az, hogy melyik földrajzi helyről (állomásról) van szó, míg az időt idővonalak reprezentálják, amelyek egyes állomásokhoz (földrajzi helyekhez) tartoznak. Az idővonalak az indulási és érkezési időpontokat tartalmazzák. Minden egyes állomáshoz tartozik egy idővonal, és az állomás minden lehetséges indulási és érkezési időpontjához egy csúcspontot definiálunk annak idővonalán. (Ha több, különböző járat indulási, vagy érkezési időpontja egybeesik, azokhoz egy „összevont” csúcspont tartozik.) Könnyű észrevenni, hogy a két modell közötti alapvető különbség az, hogy az időpontok itt helyekhez vannak kötve. Így a hálózat N csúcshalmazát az állomásokhoz tartozó indulási és érkezési időpontok adják. Minden $d \in D$ depó esetén hasonlóan definiálhatjuk E_d-t, mint a kapcsolatalapú modellben. Természetesen a depókhoz is idővonalakat adunk meg, így hasonlóan definiálható R_d is.

A legfőbb különbség azonban a két modell között a rezsijáratok definíciójában rejlik. Az idő-tér hálózati modellben ugyanis az idővonalak használatával lehetőség nyílik arra, hogy „összegyűjtsük”, összevonjuk több rezsijárat lehetséges folyamát. Így nem feltétlenül szükséges minden egyes lehetséges rezsijáratot külön éllel reprezentálni, hanem megfelelő rezsiátmenetekhez tartozó éleket összevonhatunk egyetlen éllé. A szerzők [48]-ban az úgynevezett utolsó-első egyezési stratégiát alkalmazták. Ennek alapgondolata egy kétfázisú összevonási stratégia:



6. ábra. Egy egyszerű példa idő-tér hálózatra [10].

- Az első fázisban minden egyes járat érkezését jelképező csúcspontból az összes többi állomás esetén csak a másik állomás első olyan járatához vezető, rezsijáratot jelentő élt húzzuk be, amely járatunkkal időben az első kompatibilis járat a másik állomáson. Ezen éleket nevezzük első egyezésnek. Csak ezeket az éleket tekintjük a lehetséges rezsiélek közül.
- Az első fázis után egy adott állomásnak az indulást jelképező csúcspontjaiba mindegyik másik állomásról érkező rezsijáratok közül már csak ez első egyezésnek megfelelő rezsijáratokat hagytuk meg. A második fázisban tovább csökkentjük ezen élek számát. Most kerül sor arra, hogy összevonjuk a beérkező rezsiéleket. A második fázisban egy adott állomás időpontjaihoz végignézzük az összes többi állomásról érkező, első egyezés éleket, és az oda egy azonos, de másik állomásról érkező, első egyezést jelentő élek közül csak a legkésőbb indulót hagyjuk meg. Ezt nevezzük utolsó-első egyezést jelképező élnek. Elhagyva a többi, első egyezést jelképező élet, az élek számának további csökkentése lehetséges.

Ezért a B_d halmaz az összes rezsijáratnak csak egy részét fogja tartalmazni. Ezen a módon csökkenthető a különböző állomások között rezsiélek száma. Azonban, hogy teljessé tegyük a modellt, ahhoz mindegyik állomáshoz be kell vezetni az állomáson belüli csúcspontokat összekötő, úgynevezett várakozó élek W_d halmazát, minden $d \in D$ depó esetén. Ezek az élek mindig az állomás idővonalát követik, éllel összekötve az egymást követő (indulási) időpontokat, összegyűjtve azok folyamát. Ebben az esetben az A_d élhalmaz definíciója a következő lesz:

$$A_d = E_d \cup B_d \cup R_d \cup K_d \cup W_d, \quad \forall d \in D,$$

és a gráf összes éleinek halmaza

$$E = \cup_{d \in D} A_d.$$

A 6. ábra mutat egy egyszerű példát egy idő-tér hálózatra.

Ebben az esetben az IP-modell a kapcsolatlapú modelléhez hasonló lesz. Az egyetlen különbség, hogy több él folyamát egyetlen élbe gyűjthetjük össze, így az $x_e^d \in \{0,1\}$ feltétel helyett az „$x_e^d \geq 0$, x_e^d egész” feltételt kell szerepeltetnünk.

2.3.3. A halmazpartícionálási modell

Ebben az alfejezetben Ribeiro és Soumis [63] cikke alapján megadjuk a probléma egy halmazpartícionálási megfogalmazását. Az MDVSP a 2.3.1. alfejezetben megfogalmazott G gráfon megadott körfolyamok segítségével újrafogalmazható. Minden $d \in D$ esetén legyen H_d azon G-beli p utak halmaza, amelyek $dt(d)$-ből indulnak, és oda is térnek vissza. Legyen

$$H = \bigcup_{d \in D} H_d$$

az összes G-beli ilyen utak halmaza. Minden $p \in H_d$ esetén vezessünk be egy y_p^d bináris változót, amelyet a következőképpen definiálunk:

$$y_p^d = \begin{cases} 1, & \text{ha a } p \in H_d \text{ út szerepel a megoldásban,} \\ 0, & \text{különben.} \end{cases}$$

Definiáljuk továbbá $a_{e,p}^d$-t az alábbi módon:

$$a_{e,p}^d = \begin{cases} 1, & \text{ha a } p \in H_d \text{ út tartalmazza az } e \text{ élt,} \\ 0, & \text{különben.} \end{cases}$$

Legyen c_p a $p \in H_d$ úthoz rendelt költség. Ekkor a modell a következőképpen fogalmazható meg. Minimalizáljuk a

$$\sum_{d \in D} \sum_{p \in H_d} c_p y_p^d$$

célfüggvényt a

$$\sum_{d \in D} \sum_{p \in H_d, e=(dt(u),at(u)) \in E_d} a_{e,p}^d y_p^d = 1, \quad \forall u \in U$$

és

$$y_p^d \in \{0,1\}, \quad \forall d \in D, \forall p \in H_d$$

feltételek mellett.

Ha a d depóban korlátos, k_d számú jármű van, akkor ebben az esetben a feltételeket ki kell egészíteni a következő egyenlőtlenségekkel:

$$\sum_{p \in H_d} y_p^d \leq k_d, \quad \forall d \in D.$$

2.3.4. Heurisztikus algoritmusok

Az MDVSP-re számos heurisztikus megközelítést publikáltak. Kliewer és társai egy ún. *változófixálási (variable fixing) heurisztikát* adnak meg [47], amely az idő-tér modellen alapul. Ennek alapötlete, hogy egyszerűsített problémák megoldásával (az említett cikkben külön SDVSP-problémákat oldanak meg az eredeti MDVSP minden depójára) olyan járatsorozatokat találjon, melyek minden egyszerűsített problémában egymás után következnek. Ezeket a sorozatokat leköti, és a felépítendő idő-tér modellben együtt kezeli őket.

Suhl és szerzőtársai [69] egy kerekítéses heurisztikát alkalmaztak. Módszerük alapötlete, hogy az MDVSP-feladat LP-relaxációjának értéke és az optimális egész megoldás értéke közötti különbség meglehetősen kicsi, néha nulla, és az LP-relaxáció optimális megoldásában sok változó kap már egész értéket. Az algoritmus egy előre meghatározott korlát elérésekor leállítja az IP-megoldót (ez lehet a bejárt csúcsok számának korlátja, vagy az aktuális feladat értéke és az LP-relaxáció értéke közötti különbség), majd az így „kézben lévő" részmegoldáson végrehajt egy kerekítési algoritmust. Az algoritmus által meghatározott két kerekítési tartomány $[0, r_l]$ és $[r_u, 1]$, ahol $0 \leq r_l \leq r_u \leq 1$. Legyen x_j egy változó, és $x_j - \lfloor x_j \rfloor = f_j$, ahol $0 \leq x_j \leq 1$. A heurisztika az x_j értékét az alábbi szabályok szerint kerekíti:

$$x_j := \begin{cases} \lfloor x_j \rfloor, & \text{ha} \quad f_j \in [0, r_l], \\ \lceil x_j \rceil, & \text{ha} \quad f_j \in [r_u, 1]. \end{cases}$$

Egymás után több kerekítési iterációt is végrehajthatunk, feltéve, hogy az így létrejött LP még mindig lehetséges megoldást ad. Ha nem tudtunk egy változót sem kerekíteni, akkor a kezdeti kerekítési intervallumok növelhetők. Az így kapott feladatra újra lefuttatjuk a korlátozás-szétválasztás módszerére alapuló IP-megoldót. A folyamatot addig ismételjük, amíg minden változót fixáltunk, vagy a feladatnak nincs lehetséges megoldása.

Pepin és szerzőtársainak dolgozata több heurisztikus módszert hasonlít össze [61]. A publikációban öt különböző heurisztikus megközelítést vizsgálnak:

- egy szétválasztás és vágás (*branch-and-cut*) típusú megoldást,
- Lagrange-relaxációra alapuló heurisztikát,
- oszlopgenerálást,
- egy nagy szomszédsági keresést,
- valamint egy tabu-keresést.

A szétválasztás és vágás típusú megoldás alkalmazásánál a feladat idő-tér modelljét építik fel, majd oldják meg CPLEX-szel, míg az oszlopgenerálásnál lényegében szintén CPLEX-szel kapnak eredményt.

A probléma folyammegmaradási feltételének Lagrange-relaxálásával a kapott feladat SDVSP-részproblémákkal lesz ekvivalens, amit egy aukciós algoritmussal oldanak meg. Az így kapott alsó korlátot szubgradiens módszerrel javítják.

A nagy szomszédsági keresés egy kezdeti megoldásból kiindulva minden iterációban r különböző járműműszakot választ ki, több választási stratégia valamelyikével. Az így kiválasztott műszakokat újraoptimalizálják, az oszlopgenerálást használva. A választási stratégiák valószínűsége minden iterációban változik attól függően, hogy az előzőek mennyire voltak hatékonyak.

A tabukeresés szintén egy kezdeti megoldásból indul, és minden iterációban az aktuális megoldás egy szomszédjára tér át. Kétféle módon definiálják a szomszédságot: 1-mozgatás, és csere-mozgatás alapján. Az 1-mozgatással olyan szomszédok érhetőek el, melynél a v_i eszköz valamely járatát a szomszédban egy másik, v_j eszköz hajtja végre. A csere-mozgatás olyan szomszédokat ad, melyeknél, ha a t_k járatot a v_i eszköz, valamint a $t_{k'}$ járatot a v_j eszköz hajtotta végre az eredeti ütemezésben, úgy a szomszédban a t_k járatot a v_j, valamint a $t_{k'}$ járatot a v_i eszköz hajtja végre (valamely k, k', i és j értékekre).

3. A járművezető-ütemezési feladat

A munkaerő költsége egy nagyon fontos, kiemelt összetevője az egész operatív ütemezés költségének, így a dolgozók munkájának beosztása, ütemezése manapság központi kérdés. Tipikus személyzetbeosztási alkalmazások jelennek meg a kórházak üzemeltetésénél (pl. nővérek), a call-centerekben (operátorok), légitársaságoknál (stewardessek, pilóták), és a közlekedési vállalatoknál (járművezetők munkájának ütemezésére). A beosztások elkészítésekor a feltételek erősen szakmaspecifikusak, így a problémák megfogalmazása, a modellek és megoldási módszerek is meglehetősen változatosak [28].

A járművezető-ütemezés során (amelyet röviden vezetőütemezésnek is fogunk nevezni) adott az ellátandó, operatív feladatok halmaza. Ezt kell olyan módon műszakokba rendezni az adott időperiódusra (általában egy napra) vonatkozóan, hogy minden feladat hozzá legyen rendelve egy műszakhoz, és a műszakok minden – a járművezetők számára előírt – szabálynak megfeleljenek. A hozzárendelést úgy kell megadni, hogy eközben minimalizáljuk az ütemezés során kialakított műszakok költségeinek összegét.

Sok olyan feltétel van, amelyet a vonatkozó jogszabályok és az adott közlekedési vállalat dolgozóira vonatkozó szabályok, előírások határoznak meg. Ilyenek a napi munkaidő-beosztásra vonatkozó szabályok, a maximális munkaidő és a napi pihenőidő minimális hossza, a kellő számú és idejű szünet előírása egy adott vezetési idő után, két munkabeosztás között kötelezően előírt pihenőidő stb. A menetrendi járatokon és a rezsimeneteken kívül sokféle technikai és adminisztrációs feladat is van, amelyeknek pontosan meghatározott időbeli hossza van. Ilyenek az utasok ki-és beszállási ideje a járatok végállomásain: a tankolás, a parkolás, a műszakkezdet és -befejezés stb. Megjegyezzük, hogy a vezetési idő és a munkaidő különböző fogalmak, így ezekre eltérő szabályok vonatkozhatnak, miként arra is, hogy melyik technikai tevékenység ideje melyikbe számít bele. Ehhez járulhat még hozzá az, hogy különböző vállalatoknál további korlátozó feltételek és szabályok is előfordulhatnak (szolgálati idő, tengelyen töltött idő stb.).

A különböző tevékenységekhez tartozó dolgozói költségeknek (a fizetéseknek és egyéb járulékoknak) megfelelően az egyes feladatokhoz költségeket tudunk rendelni. Így az egyes műszakok költségeit úgy számítjuk ki, hogy összeadjuk a különböző tevékenységek költségeit.

A legnépszerűbb megközelítés a probléma megoldására a halmazpartícionálás (lásd pl. [30]), valamint az annak relaxációját jelentő halmazlefedési modell (pl. [66, 73]) vizsgálata.

A megoldás előállítása tipikusan kétféle módon történhet: vagy a korlátozó feltételeket kielégítő egy lehetséges megoldás előállításával [27] és annak iteratívan történő javításával, vagy nagyszámú legenerált lehetséges megoldás közül a legjobbak kiválasztásával [50]. Különböző jellegű algoritmusokat alkalmaztak a probléma megoldására. Ezek közül érdemes kiemelni az egészértékű programozási modelleket (lásd pl. [72]), az evolúciós elvekre épülő metaheurisztikus eljárást [50], vagy a Fuzzy-módszeren alapuló [52] algoritmusokat.

A helyes modell kialakítását nagymértékben nehezíti az, hogy a különböző közlekedési társaságok más-más elveket, belső szabályokat alkalmaznak a vezetőütemezésnél. Ilyen lehet például annak előírása, hogy mely pontokon (mely földrajzi helyeken) történhet vezetőcsere egy adott járművön, ez mennyi időt igényel, milyen szabályok vonatkoznak a szünetekre stb. Az ilyen különbségek és a nagyszámú, nehéz korlátozó feltétel miatt a legtöbb megoldási módszer csak a legalapvetőbb korlátozó feltételeket veszik figyelembe. Ilyen lehet például a maximális munkaidő, a néhány rövid és egy hosszú (étkezési) munkaközi szünet.

Egy valós életből származó példa esetén az összes szünet kezelése, ütemezése azonban nem egyszerű: az egy műszakon belül előírt szünetek száma ugyanis függhet a műszak és/vagy a már ledolgozott munkaidő hosszától. A munkaidő előrehaladásával egyre gyakrabban kell a szüneteket kiadni (általában egyre rövidebb munkaszakaszok után, de ezekre is vonatkozhatnak bonyolultabb szabályok); több különböző variáció van a szünetek hosszára. Emellett sok szabály nem a munkaidőre vonatkozik, hanem a vezetési időre, és a kettő eltérhet egymástól. A munkaidőbe beleszámolódhatnak más, előírt technikai idők is, például a végállomásokon történő fel- és leszállási idők [29]. A fenti korlátozó feltételeket azért soroltuk fel, hogy érzékeltessük azt, hogy mennyire bonyolult egy ilyen rendszer, és próbáltuk felvázolni azt is, hogy milyen elvárásokat támasztanak egy matematikai modell elkészítése és alkalmazása során az egyes felhasználók.

A vezetőütemezés központi logisztikai probléma a tömegközlekedésben, hiszen a járművezetőkkel kapcsolatos költségek a teljes, közlekedéssel kapcsolatos költségek nagy részét alkotják. A vezetőütemezés alapproblémája a következő módon határozható meg: Adottak a járművek közlekedései/tevékenységei, fix indulási és érkezési idejükkel és helyükkel. A feladat az, hogy mindegyikhez járművezetőket rendeljünk minimális költséggel, átfedések nélkül, kielégítve a szabályokat és az előírásokat. Ennek klasszikus matematikai megfogalmazása egy halmazlefedési problémát eredményez, amely problémáról ismert, hogy NP-teljes [37].

A vezetőütemezési problémára *CSP-ként (Crew Scheduling Problem)* fogunk hivatkozni, bár az angol nyelvű szakirodalomban emellett szokás Driver *Scheduling Problem*-nek vagy *Duty Scheduling Problem*-nek is nevezni. Az irodalomban számos CSP-megoldási módszer és alkalmazás található. A következőkben ezeket tekintjük át röviden. Részletesebb áttekintéshez például a [28] tanulmány ajánlható az ez iránt érdeklődőknek.

3.1. Modellek és algoritmusok a CSP-re

Az alfejezetet az alapvető jelölések összefoglalásával kezdjük. Egy olyan földrajzi helyet, ahol egy *feladat* (*task*) kezdődik, vagy befejeződik, és a vezető elhagyhatja, vagy elfoglalhatja a járművet, *váltási helynek* (*relief point*) nevezünk. Amikor egy jármű egy váltási helyhez ér, *lehetséges váltási pontról* (*relief oppurtunity*) beszélünk. A jármű két egymást követő lehetséges váltási pont közötti feladatait *munkaszakasznak* (*work piece*) nevezzük. A vezetőütemezési feladat megoldására két általános megközelítés létezik.

3.1.1. A generálás és kiválasztás módszere

A módszerre az angol elnevezés (Generate and Select) rövidítése alapján GaS-sel fogunk hivatkozni. GaS-technikát használtak az [50, 52, 72] cikkekben. A módszer a következőképpen foglalható össze: A generálási lépésben állítsunk elő nagyszámú szabályos műszakot, majd a kiválasztási lépésben keressünk egy olyan részhalmazt a generált műszakokból, amely minimális költségű és lefedi a feladatokat.

Az első fázis jelentős számítási időt igényel. A számítási igény nagyban függ a járatok mennyiségétől, a szabályok számától és bonyolultságától. Emellett a szabályok ellenőrzésének számítási igénye is nagyban befolyásolja ennek a fázisnak – és így az egész probléma – komplexitását.

A GaS kiválasztási fázisa hagyományosan egy halmazlefedési vagy halmazpartícionálási feladatként fogalmazható meg. Jelölje n' a lehetséges műszakok és m' a feladatok számát, $x_j \in \{0,1\}$ és $a_{i,j} \in \{0,1\}$ az i. feladathoz és a j. műszakhoz tartozó változókat ($i = 1,2,\dots,m'$, $j = 1,2,\dots,n'$), ahol

$$x_j = \begin{cases} 1, & \text{ha a } j. \text{ műszakot kiválasztjuk,} \\ 0, & \text{különben.} \end{cases}$$

$$a_{i,j} = \begin{cases} 1, & \text{ha az } i. \text{ feladatot tartalmazza a } j. \text{ műszak,} \\ 0, & \text{különben.} \end{cases}$$

Ekkor a feladat a következő módon írható fel:

$$\min \left(w_1 \sum_{j=1}^{n'} c_j x_j + w_2 \sum_{j=1}^{n'} x_j \right) \tag{1}$$

figyelembe véve az alábbi korlátozó feltételeket:

$$\sum_{j=1}^{n'} a_{i,j} x_j \geq 1, \quad i = 1, 2, \ldots, m', \tag{2}$$

ahol c_j a j. műszak költsége, w_1 és w_2 különböző súlyok.

Az (1) célfüggvény a (2) feltételrendszerrel egy halmazlefedési feladatot generál, melynél – mint az észrevehető – átfedés is lehetséges. Ez azt jelenti, hogy ugyanazon feladathoz elvileg több műszak is hozzárendelhető. Ebben az esetben a gyakorlatban gondoskodnunk kell az esetleges átfedések kezeléséről, vagy ezek számának minimalizálásáról is [25, 66].

A partícionálási modell a lefedési feladat olyan megszorítása, amikor átfedés nem lehetséges. Ez annyiban módosítja a feltételrendszert, hogy az egyenlőtlenségek helyett egyenlőségeket írunk elő. A probléma az, hogy ebben az esetben a lehetséges megoldások létezése nem garantált. Megjegyezzük, hogy mindkét feladatról bebizonyították, hogy NP-nehéz [37].

A probléma megoldására számos eljárás létezik. Például egészértékű programozási módszereket használnak [72]-ben, heurisztikus megoldási technikákat alkalmaznak [50]-ben.

3.1.2. A konstruktív megközelítés módszere

A konstruktív megközelítés egyetlen megoldást épít fel irányítottan egy optimalizálási eljárás segítségével. Ennek kerete rendszerint egy hagyományos iteratív eljárás, amely egy induló megoldást generál, és iteratívan javítja azt. Ezzel a módszerrel találkozhatunk a [2, 27, 40, 71] cikkekben.

3.1.3. A szabályok ellenőrzése

A legkritikusabb részfeladat annak ellenőrzése, hogy a kapott megoldás lehetséges megoldás-e. Ez azt jelenti, hogy – mindkét megközelítés generáló folyamata során – a megoldáshoz tartozó összes műszakról el kell döntenünk, hogy azok teljesítik-e a feltételeket. Miután – még egy kisméretű probléma esetén is – a lehetséges műszakok száma nagyon nagy, ezért a technikák többségénél különböző egyszerűsítések kerülnek végrehajtásra azért, hogy a futási időt csökkentsék. Egy alkalmazott megoldás az, hogy a probléma méretét csökkentik azzal, hogy lehetséges váltási pontokat hagynak el, vagy azzal, hogy csak számításilag kezelhető számú műszakot generálnak le. Ezek az egyszerűsítések természetesen befolyásolhatják és korlátozhatják a módszernél az optimalizálás sikerét.

3.2. A CSP korlátozó feltételeiről

A CSP megoldása során a korlátozó feltételek valóban döntő szerepet játszanak a megoldási módszer szempontjából. Ezek erősen befolyásolják a megoldás milyenségét és a számítási időt. Erős korlátozó feltételekkel rendelkező valós problémák esetén nehéz kérdés, hogy megtalálják az egyensúlyt a fenti szempontok között. Mivel általában a probléma mérete nagy, és az előírt szabályok rögzítettek, a fő cél az, hogy találjunk egy hatékony módszert, amely gyakorlati szempontból jó minőségű, kivitelezhető megoldást nyújt.

A fő korlátokat nemzetközi (pl. EU-s szabályok) és nemzeti (minisztériumi, önkormányzati stb.) szinten határozzák meg. A helyi közlekedési vállalat is további „kemény” és „puha” igényeket (szigorúan betartandó vagy ajánlott) definiálhat. Fontos igény a valós, gyakorlati alkalmazások mellett a rugalmasság. Sok esetben az adaptálhatóság és a futási idő a legfontosabb szempontok, ha a megoldás minősége megfelel bizonyos követelményeknek (általában valamilyen korlátozó feltételeknek). Abban az esetben, ha döntéstámogató az ütemezés, egyes mérnöki követelmények minőségére vonatkozóan a megoldást nem kell, vagy nem is lehet formalizálni. Ekkor a hosszútávú tervezés ütemezése interaktív mechanizmusként valósul meg, azaz interaktív, mérnöki beavatkozás lehetséges vagy szükséges is. Egy módszer alkalmazásának lehetősége nagymértékben függ attól, hogy miként lehet a modellben a szükséges szabályokat megfogalmazni, formalizálni, milyen paramétereket képes kezelni, és ezáltal a különböző megoldásokat előállítani.

Ez a rugalmasság központi kérdés az optimalizálási folyamat többi szakaszához való viszony tekintetében is: a megoldás értékelése önmagában nem, csak a járműütemezés, valamint a vezetőbeosztás eredményének értékelésével együtt lehetséges.

3.3. A célfüggvényről

A CSP megoldásának minősége csak részben függ az eredményezett műszakok számától. Általánosabban, egy megoldás költségének a tartalmazott műszakok költségeinek összegét tekinthetjük. Természetesen egy járművezető foglalkoztatásának van egy általános költsége, ezért minimalizálva az összköltséget, a vezetőütemezés alacsony szinten fogja tartani a műszakok számát is. Másrészt, költségeket rendelhetünk a műszakokhoz a bennük szereplő feladatok szerint is. Ezen kívül meg lehet határozni egyéb jellemzőket, mint például a műszak típusa, a várható időtartama, amelyek járulékosan, büntetésként szerepelhetnek a költségben is. Mindazonáltal, a legáltalánosabb értelemben értékelve a megoldást, a műszakok összes munkaidejét lehet teljes költségként tekinteni. (Egyes helyeken a bérezés nem munkaidő szerint történik, ilyenkor természetesen a műszak költségét is ehhez kell igazítani.)

4. Integrált stratégiák

Ebben a fejezetben a jármű- és járművezető-ütemezési feladat *integrált megközelítéseit* (*vehicle and crew scheduling problem, VCSP*) tekintjük át röviden, megemlítve az irodalomban tárgyalt legfontosabb modelleket, algoritmusokat. Az érdeklődőknek kiindulópontként a [68] cikket ajánljuk, a részletesebb elmélyüléshez javasoljuk a kitűnő áttekintést adó [24] tanulmányt.

A járművezető-ütemezés a hagyományos megközelítést követve a járműütemezés fázisa után hajtódik végre. (Ezért ezt *szekvenciális megközelítésnek* is nevezzük.) Ez általában hatékonyan végrehajtható, ha a váltási helyek között sok olyan van, amely megengedett váltási pont is egyben, és a kialakított járműütemezés gyakran érint ilyen váltási helyet. Azonban, ha a kialakított járműütemezés túl „sűrű", sok helyen nincs elég idő a járművezető-váltásra, vagy a kialakított járműütemezés hosszú ideig nem érint megengedett váltási helyet, akkor ronthatunk az előző fázis eredményén, veszíthetünk a hatékonyságból. A járműütemezés és a járművezető-ütemezés egyszerre történő elvégzése emiatt indokolt lehet, és ez napjaink egyik fontos kutatási területe.

A VCSP feladata a következőképpen fogalmazható meg. Adott menetrendi járatok egy halmaza, adott a járműflotta, melynek járművei különböző depókhoz tartoznak, adottak továbbá a járművezetőkre vonatkozó szabályok. A feladat a járműveknek és a járművezetőknek olyan érvényes ütemezését adni, hogy az valamennyi – járműre és járművezetőre – vonatkozó szabálynak megfeleljen, és minimális költségű legyen.

Ez a feladat egy, az MDVSP-hez hasonló egészértékű programozási feladatként írható fel, kibővítve olyan feltételekkel, amelyek egyrészt a vezetőütemezésnek felelnek meg, másrészt pedig kapcsolatot teremtenek a két ütemezés között. Ekkor a célfüggvényben is a két ütemezés költségének összege jelenik meg (lásd pl. [24]).

Mint korábban láttuk, az MDVSP- és a VCSP-feladat kapcsán is elmondhatjuk, hogy mindkettő NP-nehéz feladat.

A következőekben ismertetett módszerek esetén néhánynál a járműütemezés részfeladata egydepós, és így polinomiális időben megoldható [13].

A VCSP feladatában a lehetséges műszakok száma igen nagy, különösen a többdepós esetben. Éppen ezért az 1990-es évek végéig nem övezte akkora érdeklődés a problémát a kutatók körében, mint a járműütemezési vagy járművezető-ütemezési problémát. A számítógépek számítási sebességének növekedése és az egyre kifinomultabb optimalizálási módszerek alkalmazása révén azonban az utolsó évtizedben megnőtt az érdeklődés ezek irányában, szaporodnak az ilyen tárgyú publikációk. A kis és közepes méretű problémák mára megoldhatóvá váltak.

Már az 1980-as évek elején Bodin és szerzőtársai komolyan kritizálták a szekvenciális megközelítést [13]. Észak-amerikai tömegközlekedési példákon keresztül bizonyították azon érvelésüket, hogy a járművezetők (bér)költsége sokszor magasabb, mint a járművekkel kapcsolatos költségek. Extrém esetekben a bérköltség akár 80%-át is adhatja a kétféle költség összegének. Ebből az következik, hogy ezt a költséget nem lehet másodlagos kérdésnek tekinteni. Sőt, alapvetően figyelembe kell venni már az első fázisban, vagy kombinált, integrált módon. Ball és szerzőtársai [8] ugyancsak integrált modellt javasoltak.

Az ezt követő cikkekben különböző heurisztikus módszereket adtak meg a kutatók, amelyekben a VCSP heurisztikus megoldása során figyelembe vettek bizonyos, járművekre vonatkozó feltételeket is. Az 1997 előtt használt módszerekről egy jó áttekintés található Gaffi és Nonato 1999-es közleményében [35], vagy Freling 1997-es PhD-dolgozatában [32].

Az első, valóban integrált jármű- és vezetőütemezési megközelítés csak 1995-ben született Freling és szerzőtársai révén [31]. Erről Gaffi és Nonato [35] bizonyította be, hogy hatékony lehet akkor is, amikor a megengedett váltási pontok távol esnek egymástól, például helyközi vagy távolsági tömegközlekedés esetén. A módszer szintén jól működhet azokban az esetekben, amikor egy vezető egy járművet használhat csak a műszakja során.

A legnépszerűbb megközelítés az integrált VCSP-problémára kétségkívül az egészértékű programozási modell alkalmazása. Freling és szerzőtársai [31] modellje az egydepós esetre három részből állt: az SDVSP-t kvázihozzárendelési, a VCSP-t halmazpartícionálási feladatként fogalmazták meg. A kettőt korlátozó feltételekkel kötötték össze, biztosítva a kompatibilitást. A feladat megoldására Lagrange-relaxációt és oszlopgenerálási módszert kombináló, közelítő megoldást adtak. Ez aztán további publikációkat inspirált, lásd [33, 34, 46].

Freling és szerzőtársai a [33, 34] cikkeikben szintén az egydepós integrált feladatot tekintették, a buszok számára vonatkozó felső korlát nélkül. Ugyancsak oszlopgenerálást – és ezen belül Lagrange-relaxációt – használtak az ott adódó ún. mesterprobléma megoldására, a feltételek relaxációjával. A [34] cikkben valós, kb. 150–250 járatból álló feladatok megoldásáról számoltak be, kezelhető megoldási időben. Megmutatták, hogy csak kis javítás érhető el a feladatok szekvenciális, egymás után történő megoldásához képest. Ez a javítás szignifikánsabb, ha a járművezetők nem válthatnak járművet (egy szünetet követően).

Az első pontos megoldást adó algoritmust 1999-ben publikálta Haase és Friberg [41] az egydepós esetre. Modelljükben mindkét halmazpartícionálási megközelítés integrált matematikai megfogalmazását adták: mindkét részfeladatot halmazpartícionálási feladatként fogalmazták meg. Eljárásukban a járművek ütemezése Ribeiro és Soumis 1994-es [63], míg a járművezetők ütemezése Desrochers és Soumis 1989-es [26] modelljein alapult. A szétválasztás, vágás és árazás (branch-and-cut-and-price) típusú algoritmusukban oszlopgenerálást és vágásgenerálást alkalmaztak. Az oszlopgenerálási mesterprobléma az LP-relaxációnak felelt meg, míg az árazásra (pricing) a legrövidebb út problémák megoldása szolgált. Algoritmusukkal csak kisméretű példákat tudtak megoldani, legfeljebb 20 járatból állókat.

Egy másik érdekes egzakt algoritmust adtak meg az egydepós esetre Haase és szerzőtársai 2001-ben [42]. A járművezető-ütemezési problémát egy többtermékes hálózati folyam problémaként megfogalmazva oldották meg úgy, hogy beleágyazták az egydepós, járművekre vonatkozó feltételeket. Ezt olyan módon tették, hogy mindkét fázist figyelembe véve, a VCSP-feladatra garantált, pontos optimumot adott az algoritmusuk. Itt a szétválasztás és árazás (branch-and-price) típusú algoritmusra számos gyorsítási technikát alkalmaztak. Két algoritmusváltozatot is tárgyaltak a szétválasztáskor adódó ágak kezelésére: egy egzakt és egy heurisztikus változatot. Véletlen feladatokon végeztek számítógépes szimulációs teszteket. Átlagosan kb. 1,5 órás (bár 10-ből 6 esetben 3 órás) futási idővel tudtak 150 járatméretű példákig pontos megoldást szolgáltatni. Ennél nagyobb, 350 járatméretű példákig heurisztikus megoldást alkalmaztak, 2 órán belüli CPU-időt használva. Itt a feladat olyan egyszerűsített megfogalmazását tekintették, ahol a költségben csak a szükséges buszok száma szerepelt, és ők sem vettek figyelembe korlátot ezek számára. Így megfogalmazásuk gyakorlatilag szintén egydepós feladatként fogható fel. Fontos kiemelni ugyanakkor, hogy a feladatok nem valós, hanem a tömegközlekedési feladatokat szimuláló véletlen adatokból származtak.

A többdepós esetben Gaffi és szerzőtársa [35] tárgyalták először az integrált feladatot, heurisztikus módszert használva. Lagrange-relaxációt és az oszlopgenerálás módszerét használták, csak a többdepós esetre módosítva a megfogalmazást. Ők is az egyidejű vezető- és járműütemezés előnyei mellett érveltek olasz tömegközlekedési példákon keresztül, amelyek nem városi (nem helyi) közlekedési példák voltak. De újra felhívja a figyelmünket a számítási idő fontosságára, kritikus voltára, hogy egy 257 járatból álló, olasz tömegközlekedési példa esete 24 óránál hosszabb számítási időt adott, átlagosan 2-6 óra volt a futási idő, bár a maiaknál lényegesen lassabb, 180 MHz-es PC gépen.

Huisman és szerzőtársai 2005-ben [46] a korábbi egydepós [34, 42] modelleket és algoritmusokat sikeresen terjesztették ki a többdepós problémára. Ez volt a többdepós probléma első általános matematikai megfogalmazása. Két megközelítést is javasoltak: az egyik a [32, 34], a másik a [42] tanulmányokban ismertetett módszer általánosítása a többdepós esetre. Mindkettő esetén az első fázisban egy alsó korlátot számítanak ki az optimumra. Az első fázis LP-relaxácón alapszik, oszlopgenerálást és Lagrange-relaxációt használ. A második fázis ad egy lehetséges egészértékű megoldást a feladatra. A [34]-ben használt Lagrange-relaxációt alkalmazva először egy járműütemezést generál, amelyből aztán a [33, 34] cikkekben tárgyalt módszereket használva járművezető-ütemezést készít. Összehasonlító teszteket kb. 650 járatig készítettek, amelyek eredményei felülmúlják a szokásos szekvenciális ütemezést használó módszereket. Teszteket végeztek valós és szimulációs tesztadatokon egyaránt. A két megközelítés között nem mutatkozott szignifikáns eltérés, de az első valamennyivel jobbnak bizonyult.

Valós, nagyméretű, többdepós, heterogén járműflotta esetén az eddig megemlített módszerek integrálása nehéz lenne egy alkalmazási rendszerbe. Így ezek helyett a szekvenciális, esetleg kézi módon történő integrálást alkalmazták a 2000-es évek elejéig.

A [22] cikkben rövidebb megoldási időt tudtak produkálni és nagyobb feladatokat tudtak megoldani a szerzők, kisebb méretű részfeladatokra vágva a feladatokat, és azokat – önmagukban, integrált módon – külön oldva meg. A többdepós feladatra Borndörfer és szerzőtársai [15] is Lagrange-relaxáción alapuló megközelítést használtak. A járműütemezési és a járművezető-ütemezési problémáknál használt közelítő megoldásokat alkalmazták, ezek információit felhasználva egy branch-and-bound típusú algoritmusban. Egészértékű programozási technikát használtak, különböző korlátozó feltételekkel összekapcsolva a járműveket és járművezetőket a rezsi és a kiállási, valamint beállási járatoknál. Heurisztikus módon, Lagrange-relaxációt használva, majd az egészértékű megoldásokat egy korlátozás és szétválasztás típusú technikával nyerve nagyméretű, 1500-as járatszámú problémát is kezelni tudtak.

Az integrált feladat egy heurisztikus megközelítését adja Laurent és Hao dolgozata [51] is. Habár az általuk tárgyalt feladat általánosabb, mint az egydepós eset, de feltételezik, hogy az összes jármű ugyanabban a depóban parkol. Ugyanakkor a járművek típusaik szerint nem kell, hogy homogén flottát alkossanak. Egy mohó, véletlen adaptív keresést alkalmaznak (Greedy Randomized Adaptive Search, GRASP). Hasonlóan a legtöbb módszerhez egynapos időhorizontot alkalmaznak, de a járművezetőkre vonatkozó feltételek egyszerűsítettek: csak a napi műszak „átmérőjére" (a fellépés és lelépés között eltelt időre) vonatkozó korlátot, a napi teljes munkaidőre vonatkozó korlátot, és azt a paramétert veszik figyelembe, hogy a járművezetőknek megengedett-e vagy nem a napon belüli járműváltás.

Mesquita és Paias 2008-ban [58] két matematikai megfogalmazást adtak a problémára. Mindkét modell többtermékes hálózati modellt tartalmaz a járműütemezésre, míg a járművezető-ütemezési rész vagy halmazparticionálási vagy kombinált halmazparticionálási és lefedési megfogalmazás. A relaxált LP-t oszlopgenerálással oldották meg, amelynek részproblémája korlátozó feltételekkel ellátott legrövidebb út feladat. Amennyiben a relaxált LP megoldása nem egésznek adódott, egészértékű megoldást kerestek korlátozás és szétválasztás alapú hagyományos IP-megoldóval. A [38]-ban alkalmazott eljárás egy ehhez hasonló megközelítést használt, a szerzők korábbi idő-tér hálózati modelljét alkalmazva a járműütemezési komponensben.

A részben integrált modellek közé tartozó modellt tárgyal Gintner, Kliewer és Suhl 2008-as cikke [39]. Ez annyiban hasonlít a tradicionális szekvenciális megközelítésekre, hogy a járművek optimalizált ütemezését végzi el először, majd – ennek eredményét felhasználva – a járművezetőkét. A járművek ütemezési fázisa az idő-tér hálózati MDVSP-modellt használja, de annyiban különbözik az eredeti megközelítéstől, hogy nem egyetlen optimális megoldást ad, hanem többet, minimális járműszámmal és minimális költséggel. Ezek után a VCSP-t halmazpartícionálási feladatként oldja meg, és Lagrange-relaxációt alkalmaz. A klasszikus megközelítéssel összehasonlítva a módszer jobb járművezető-ütemezéseket eredményez.

Steinzen és szerzőtársai 2010-ben [68] egy teljesen integrált VCSP-megközelítést adnak. A mögöttes, járműütemezést kezelő modell az idő-tér hálózati modell. A megoldás itt is az oszlopgenerálás és a Lagrange-relaxációs módszert kombinálva történik. Az oszlopgenerálás részfeladatát korlátozó feltételekkel ellátott – egy idő-tér hálózaton alapuló – legrövidebb út feladat megoldásával modellezték. Egy heurisztikus, szétválasztás és árazás típusú módszerrel generáltak lehetséges megoldásokat. A numerikus tesztjeik azt mutatják, hogy ez a módszer felülmúlja a korábbiakat az ismertetett tesztfeladatokon, amelyeket 640 járatot tartalmazó példaméretekig tekintettek. Az összehasonlítás alapjait a [15, 38, 45, 46] cikkek alkották, amelyek közül valamennyit felülmúlta tesztpéldáikon az algoritmusuk. Az alkalmazott tesztpéldáik (kizárólag) a Steinzen weblapján található példák [67] voltak.

5. A műszakkiosztási feladat

A járművezetők műszakkiosztási problémája is az általános műszakkiosztási feladatok közé tartozik. Itt a feladat az, hogy egy tervezési periódusban munkavállalókat – számos általános és speciális feltétel figyelembevételével – rendeljünk hozzá a napi műszakokhoz. A járművezetők műszakkiosztási feladatának feltételei általában megfelelnek más, tipikus alkalmazások feltételeinek, amelyek egyes személyzetbeosztási [17, 18] és nővérbeosztási [19, 11] feladatoknál, illetve a callcenterek üzemeltetése esetén az operátori műszakkiosztásnál [65] is megtalálhatók.

A járművezetők műszakkiosztási feladatában adottak a járművezetői műszakok, amelyeket a vezetőütemezés során meghatároztunk. Ezek napi beosztásokat jelentenek. Ezek a műszakok a tervezési periódus különböző napjain eltérőek lehetnek. Adottak továbbá a rendelkezésre álló járművezetők.

A műszakkiosztás általában hosszabb periódusra történik, amelyet *tervezési időszaknak* nevezünk. A tervezési időszak tipikusan néhány hétből álló időintervallum, de ez lehet néhány nap, hónap, vagy akár egy egész év is. A műszakkiosztási feladat lényege, hogy egy tervezési időszakra az adott műszakokat rendeljük valós alkalmazottakhoz úgy, hogy az előírt szabályokat betartjuk. Ez a feladat több alkalmazási területen fordul elő. A leggyakoribb alkalmazások a légiközlekedésben (pilóta és személyzet), vasúti közlekedésben, call-centerekben dolgozók, nővérek és buszsofőrök munkájának tervezése.

A módszernek a tervezési időszak minden műszakjához járművezetőt (konkrét személyt) kell rendelnie. Van néhány általános szabály, rendelkezés, amely korlátozó feltételként veendő figyelembe. Ilyenek például a maximális heti munkaórák száma, vagy a szabadnapok előírt minimális száma egy adott időszakban, ezen belül a vasárnapok száma, stb. A szabályok teljesen különbözhetnek az alkalmazási területtől függően (légi, vasúti közlekedés, városi tömegközlekedés stb.).

Gyakran további speciális helyi szabályok is előfordulnak, amelyeket szintén korlátozó feltételként kell figyelembe venni a beosztás elkészítéskor. Ilyen például az, hogy milyen kategóriájú vezetői jogosítvánnyal kell rendelkeznie a vezetőnek bizonyos típusú buszok vezetéséhez.

A hozzárendelés költsége a közlekedési cégek esetén nagyban függhet a vállalatvezetés „filozófiájától". Ez lehet egyetlen cél is, de több összetevő is alkothatja. Utóbbi esetben ezekből több is (pl. súlyozott módon) megjelenhet a célfüggvényben. Ilyen lehet a járművezetők számának minimalizálása, a szerződésben szereplő megállapított munkaóráktól való eltérések összegének minimalizálása (akár túlóráról, akár az abban szereplőnél kevesebb tényleges munkaidőről van szó, azaz a túlfoglalkoztatást és az alulfoglalkoztatást egyaránt kerülni kell, a lehetséges mértékben). A fenti okok miatt több célfüggvényű optimalizálási módszerek használata is indokolt lehet [59, 60]. Miután az alapfeladat általánosan tekintve hozzárendelési probléma, a szakirodalomban megtalálható néhány – a hozzárendelési feladatra alapozott – általános megközelítés is a műszakkiosztási feladatra. Ilyenek például a többtermékes folyam algoritmusok [17], a logikai programozás alkalmazása korlátozó feltételekkel [74], vagy az evolúciós módszert használó algoritmusok [59].

A feladatot több részfeladatra lehet bontani, ahol az egyes lépéseket egymás után, esetleg iteratívan lehet végrehajtani. Bizonyos környezetben nem minden lépés szükséges, illetve össze is lehet vonni őket. A buszközlekedésben leginkább előforduló lépések a következők.

– Igényfelmérés. Első lépésben meghatározható, hogy mennyi emberre van szükség. Ez függ a műszakok számától, egymáshoz képest időbeni viszonyuktól, méretüktől, típusuktól, továbbá befolyásolják a rendelkezésre álló alkalmazottak „tulajdonságai" (pl. szerződések, jogosítványok).

– Szabadnap kiosztás. A szabadnapok kiosztása főleg akkor szükséges, amikor nem teljesen kötöttek a jövőbeni műszakok. Ennek végrehajtása során némi rugalmasságot biztosítani kell. Ez a lépés természetesen a definiált szabályoktól függ, attól, hogy mennyi és milyen szabadnapot kell kiosztani.

– Műszaksorozatok készítése. Ebben a lépésben az adott műszakokból (esetleg másképp meghatározott feladatokból) bizonyos időintervallumra (jellemzően egy hétre vagy hónapra) adott hosszúságú sorozatokat hoznak létre.

– Műszaksorozat-ember hozzárendelés. Végül itt az önálló műszakokat, vagy műszaksorozatokat tényleges alkalmazottakhoz rendelik.

A feladat egy részletesebb, általánosabb felbontása megtalálható [28]-ban.

5.1. Korlátozó szabályok

A szigorú, „kemény” szabályok főleg az EU, minisztériumok, önkormányzatok, szakszervezetek stb. által előírt szabályok, ezek betartása kötelező az ütemezés során. Az ajánlott, „puha” szabályok főleg személyi igények során kerülnek előtérbe (néhány szabadnap egymás után következzen, néhány szabadnap essen hétvégére stb.). Ezek betartása nem kötelező, de minősítik a megoldás „ jóságát”, súlyozva bekerülhetnek a célfüggvénybe a kiértékelésnél.

Jellemző szigorú szabályfajták, amelyek a legtöbb alkalmazási területen előfordulnak:

– Adott az egymást követő két műszak közötti minimális pihenőidő.

– Maximálva van a heti munkaidő.

– Adott a heti minimális pihenőidő.

– Szabályozott, hogy legfeljebb hány egymást követő napon dolgozhat a munkavállaló szabadnap nélkül.

– Adott, hogy legalább hány szabadnapja legyen egy munkavállalónak egy hónapon belül.

– Adott, hogy hány szabadnapnak kell esnie bizonyos típusú napra (pl. hétvégére, vasárnapra stb.) adott időn (egy hónapon) belül.

– Adott, hogy bizonyos munkavállalók csak bizonyos típusú műszakot kaphatnak, vagy milyen típusból mennyit kaphatnak.

Jellemző „puha” szabályok, igények:

– Kerüljük az olyan kiosztást, ahol két szabadnap között csak egy munkanap van.

– Előnyben részesítjük, ha két szabadnap egymás után következik.

– Lehetőleg a szabadnapok egyenletesen legyenek elosztva a tervezési időszakban.

– Egyenletesen legyenek a műszaktípusok kiosztva az alkalmazottak között, vagy pont fordítva, lehetőleg azonos típusú műszakot kapjon egy alkalmazott egy adott időszakon belül (hét vagy hónap).

A buszos járművezetőkre vonatkozó szabályok egyfajta osztályozása megtalálható [60]-ban.

5.2. Kiértékelés

A műszakkiosztás minőségének meghatározása nem egyértelmű a szakirodalomban. Természetesen a cél mindenhol a költség minimalizálása, de a költség meghatározásához több tényezőt kell figyelembe venni. Egyrészt cél a szükséges alkalmazottak számának minimalizálása, feltételezve, hogy kevesebb ember alkalmazásával kisebb a költség [17, 18, 74]. Ugyanakkor több helyen az alkalmazottak száma kötött, ilyenkor a kiosztás minőségének növelése a cél, amit a puha szabályokhoz való illeszkedés határoz meg. Ezen szabályok megfelelően súlyozott kiértékelése határozza meg a kiosztás minőségét. Természetesen gyakran az emberek számának minimalizálása és a puha szabályok figyelembe vétele együttesen jelenik meg az optimalizálásban [19, 59], esetleg a be nem osztott alkalmazottakat tartalékba helyezik betegségek esetére [60]. Ugyanakkor a valós életben a kiosztás költségét nagyban befolyásolja az alkalmazottak szerződése, amely meghatározza a ledolgozandó órák számát. Ettől való eltérés alul-, illetve túlfoglalkoztatást jelent. Az előbbi azért költséges, mert a szerződésben foglalt órák alapján kapja a fizetését, holott nem dolgozott annyit, az utóbbi meg túlórát jelent, amelynek bérezése általában magasabb az alap órabérnél. Így ezek súlyozott minimalizálása a költség csökkenését jelenti [6]. Az nyilvánvaló, hogy az alul- és túlfoglalkoztatás csökkentése csak úgy érhető el, ha a kiosztás során felhasznált alkalmazottak száma és összetevője változik.

5.3. Megoldási módszerek

Ugyan a műszakkiosztási feladat az élet eltérő területein fordul elő, a főbb betartandó szabályok, illetve az optimalizálás során figyelembe vett költség- és célfüggvények nagyon hasonlóak. Így a különböző területen alkalmazott megoldási módszerek is elég általánosak a többi alkalmazási környezetbe való illesztéshez. A módszerek közötti különbség egyrészt inkább abból adódik, hogy hol húzzuk meg a határt a megoldás minősége, valamint a feladat mérete és a futási idő között.

Másrészt az alkalmazott optimalizálási algoritmusok változatossága jellemzi a szakirodalmat. Egyik megközelítés, hogy a feladatot visszavezetik halmazlefedési vagy partícionálási feladatra, ahol legenerálnak sok műszakkiosztást, majd ezekből kiválasztják azokat, amelyekkel a költség a legkisebb. Ezt az irányt követték például Gamache és szerzőtársai, akik 1999-es cikkükben [36] a halmazpartícionálást oszlopgenerálással oldották meg.

A feladat felírható folyamproblémaként is. Cappanera és szerzőtársai 2004-ben egy többtermékes folyamproblémaként oldottak meg légiközlekedés ütemezési feladatot, ahol a többtermékességet az adta, hogy háromféle kiosztást készítettek a kapitányoknak, pilótáknak és légi utaskísérőknek [17]. Jellemző még a kombinált megoldási módszerek használata. Yunes és társai [74] rámutattak, hogy az egészértékű programozási feladatként történő felírás csak kis feladatokra működik elfogadható időn belül, míg a pusztán korlátozó feltételes logikai programozásként való felírás már hatékonyabb, de még mindig nem elfogadható valós méretű feladatokra. Azt azonban kimutatták, hogy ennek a két módszernek a kombinálásával készített hibrid módszer, ahol oszlopgenerálást alkalmaztak és az oszlopok generálását végezték korlátozó feltételes logikai programozással, már képes volt nagy feladatokra is hatékony megoldásokat adni elfogadható idő alatt.

További kombinált módszert alkalmaztak Caprara és szerzőtársai. Itt a megoldást egy konstruktív heurisztika adja, amely a kiosztás építése közben a mohó műszakválasztáshoz felhasználja egy Lagrange-féle relaxált egészértékű programozási feladat megoldását [18]. Iteratív kombinált heurisztikát alkalmaz Bellanti [11] és Nurmi [60]. Előbbi egy kezdeti kiosztást generál mohó módon, majd ezt javítja szomszédsági keresés módszerrel, ehhez tesztelt iteratív helyi keresést, illetve tabulistás módszert [11]. Nurmi és szerzőtársai [60] a megoldást két fázisban végzik: először a szabadnapokat osztják ki, majd utána a műszakkiosztást. Mindkét feladathoz ugyanazt a módszert alkalmazzák, amely egy mohó, populációalapú helyi keresés.

Egy kiosztás kiértékelése több tényezőből áll össze, így az optimalizálás gyakran több célfüggvényű optimalizálás. Erre példa Moz és szerzőtársai [59] evolúciós módszere, ahol két szempont szerint történik az optimalizálás. Ez a kettő az emberek elvárt számától való eltérés minimalizálása és a túlórák egyenletes elosztása. A generált megoldások kiértékelésénél e két cél szerinti Pareto-dominanciát használják fel.

5.4. Szétválasztási stratégiák

A fenti részfeladatokat közelebbről megvizsgálva láthatjuk, hogy különböző opciók lehetségesek az ütemezési rendszer struktúrájának megtervezésére. Két alapvető kérdés, amely fontos ennek a struktúrának a megtervezésével kapcsolatban:

– Hol legyen a határvonal a vezetőütemezés és a műszakkiosztás feladatai között, mely szabályokat melyik részfeladatnál vegyünk figyelembe? Alapvetően a legfőbb elv ezzel kapcsolatban az, hogy a dolgozókra vonatkozó olyan szabályok, rendelkezések, amelyek az egy napon belüli műszak kialakításában veendők figyelembe, azok a vezetőütemezés, míg a további feltételek a műszakkiosztás feladatkörébe tartoznak. Sajnos, elméletileg ennek a szabálynak a figyelembevételével is a vezetőütemezéssel a napi ütemezéseknek egy olyan szerkezetét kaphatjuk, amelyekből a műszakkiosztás már nem eredményezhet a szabályoknak megfelelő megoldást. A gyakorlatban ez a probléma inkább csak kisebb városok társaságainál életszerű, ott fordulhat elő realisztikus módon (jóval százezer fő alatti lakosság esetén), mivel a kisebb kombinációs lehetőség okozhat hozzárendelési problémákat egy olyan ütemezésben, ahol az ütemezés szerkezetének előírtnak kell lennie. Mivel a feladat ebben az esetben kisebb méretű, ez lehetővé teszi napi ütemezés helyett heti periódusok kialakítását, így csökkentve annak a kockázatát, hogy nem kapunk lehetséges megoldást.

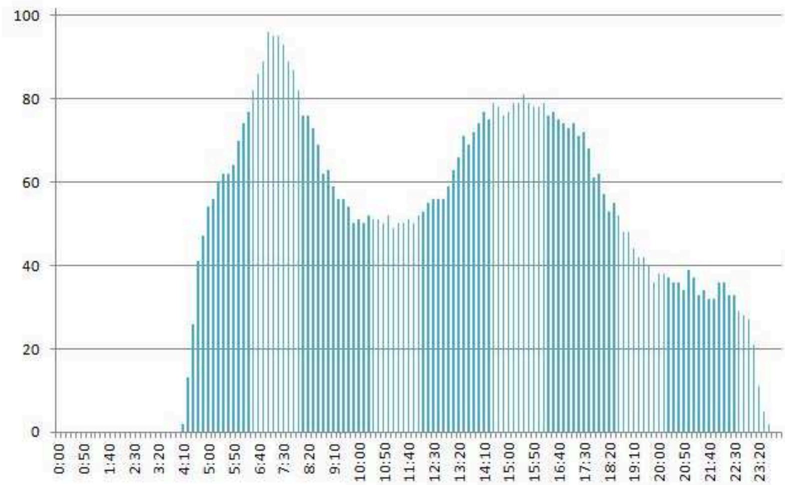
– A sorrend a járműütemezés és a járművezetők ütemezése között: mivel a járművek napi műszakjai menetrendi előírásokon alapulnak, így természetes módszerként adódik első fázisban az optimális járműütemezést elkészíteni, és azután a vezetőket hozzárendelni a járművekhez (megengedve a műszak során az eszközökön a vezetőcserét) olyan módon, hogy az összes, az emberekre vonatkozó szabály ki legyen elégítve. Ez a módszer ugyan hatékony, viszont robusztus számítógépes optimalizálási hátteret igényel. Éppen ezért a gyakorlatban, ahol nem használnak optimalizálási eszközöket automatizált tervezésre, vagy annak támogatására, az alkalmazott mérnöki „heurisztikák” megcserélik a két lépést: első lépésként (nem automatizált tervezéssel) gyakran a menetrend alapján emberműszakokat hoznak létre, figyelembe véve a járművezetőkre vonatkozó szabályokat, majd azután megfelelő járműveket rendelnek a kialakított műszakokhoz. Ennek a megközelítésnek az előnye az, hogy a feladat komplexitása jelentősen csökken, mivel a járművezető a műszakja alatt nem cserélhet járművet. Komoly hátrány ugyanakkor, hogy ez magasabb költségű ütemezéseket eredményezhet. Jellemzően, az egy napon használt járművek száma ilyenkor szignifikánsan nagyobb. A fentiek alapján egy optimalizálásorientált automatizált ütemezési rendszer esetén feltételezzük, hogy az járműütemezéssel indul, vagy ha nem, akkor ezzel egyidejűleg figyelnie kell a járműszabályokat is.

6. Néhány saját eredmény

Végül röviden megemlítjük, hogy a jelen tanulmány szerzői további (részben más társszerzőkkel közös) cikkeikben a közösségi buszközlekedés operatív tervezési feladataira adott – általában különböző gyakorlati kihívásokból fakadó – megoldásaikat tárgyalják az alábbi cikkekben, közleményekben: [4, 5, 6, 7, 9, 10, 20, 21, 70, 71].

Az itt tárgyalt felosztás alapján megvalósított, implementált fázisokat az [5] és a [10] publikációkban egy keretrendszerben mutattuk be. A gyakorlatban is egy olyan, nagy modulokból álló keretrendszernek nevezhető rendszert fejlesztettünk, amelynek moduljai megengedik olyan eljárások beágyazását, amelyek megfelelnek a fent említett követelményeknek. Amint az [10]-ben tárgyalt, ez a keretrendszer lehetőséget ad arra, hogy egy-egy moduljába különböző opciókat lehessen beépíteni. Így egy-egy modulon belül alternatív módszerek alkalmazhatók opcionális megoldásként, és néhány esetben kitérünk ezeknek a lehetőségeknek a tárgyalására és vizsgálatára is.

A járműütemezési feladatra a [20] és a [21] publikációkban az MDVSP-heurisztikák kapcsán tárgyalt, a [47] cikkben megadott változófixálási heurisztika néhány módosítását, továbbfejlesztését tárgyaltuk, különböző hasonló típusú, további heurisztikákat tárgyalva és elemezve.



7. ábra. Egy tipikus példa: egy munkanapon belül melyik időszakban hány jármű (és járművezető) szükséges minimálisan a járatok ellátásához. Ennek az ún. „kétpúpú teve” diagramnak a „púpjai”, a reggeli és a délutáni csúcs (iskola- és munkakezdési, illetve befejezési időpontok) jellegzetesek munkanapokon.

A járműütemezési és jármű-hozzárendelési feladatokhoz kapcsolódóan tapasztalatunk szerint a konkrét, gyakorlati ütemezésben a szakirodalomban tárgyalt és alkalmazott modellek hátrányaként lehet megemlíteni, hogy csak azokat a szabályokat veszik figyelembe, amelyek a menetrendi járatokkal, az azokhoz megkívánt busztípusokkal és kapacitásokkal, valamint a menetrendi járatok közötti rezsijáratokkal kapcsolatosak. Nem lehet viszont olyan specifikus feltételeket beépíteni, amelyek valós alkalmazási környezetből származnak. A tömegközlekedés operatív feladatainak tervezésénél ilyen tipikus, járműspecifikus korlátozó feltételek a tankolási vagy parkolási előírások. Tankolási követelmények beépítését tárgyaltuk [7]-ben és [9]-ben. Az ezekben tárgyalt módszer előnye, hogy heterogén járműflotta tankolási feladatait is kezeli. Erre szükség lehet, amennyiben hosszú tankolási idejű, egy tankolással a dízel üzemenyagú járművekhez képest kis távolságot megtenni képes járművek (is) vannak a flottában.

Egy járművezető-barát, azaz olyan módszert tárgyalunk [5]-ben a jármű- és vezetőütemezési feladat iteratív megoldására, ahol a járművezetőknek nem kell a nap során járművet cserélni. Így csak egyazon járművet vezetik a nap során a műszakjuk alatt, kivéve osztott műszak esetén. Ez olyan napközbeni, több órás otthoni pihenő utáni visszatérést tartalmazó műszak, amely tulajdonképpen két, független műszakrészből áll. A 7. ábra mutatja, hogy az osztott szolgálat miért szükséges általában munkanapokon: a reggeli és a délutáni csúcs járatainak ellátásához több jármű (és járművezető) szükséges, mint egyébként napközben.

A [71] publikáció a vezetőütemezésre ad a gyakorlatban használható heurisztikákat, míg [70] a vezetőütemezéshez kapcsolódó tevékenységekre egy általános keretmunkát.

A [6] publikációban a műszakkiosztási feladatra adott hosszú távú, több hetes intervallumokra adott megoldásainkat ismertetjük.

Köszönetnyilvánítás.

A kutatást a „Szuperszámítógép, a nemzeti virtuális laboratórium" című, TÁMOP-4.2.2.C-11/1/KONV-2012-0010 azonosítószámú projekt támogatta az Európai Unió és az Európai Szociális Alap társfinanszírozása mellett. A tanulmány a TÁMOP-4.2.1/B-09/1/KONV-2010-0003 Mobilitás és környezet: Járműipari, energetikai és környezeti kutatások a Közép- és Nyugat-Dunántúli Régióban projekt támogatásával jött létre, a projekt a Magyar Állam és az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

Hivatkozások

[1] E. Abbink, J. van 't Wout, and D. Huisman: *Solving Large Scale Crew Scheduling Problems by using Iterative Partitioning*, In Proceedings of ATMOS2007 – 7th Workshop on Algorithmic Approaches for Transportation Modeling, Optimization and Systems, 96–106, 2007.

[2] U. Aickelin, E.K. Burke, and J. Li: *Improved Squeaky Wheel Optimization for Driver Scheduling*, In Proceedings of the 9th International Conference on Parallel Problem Solving from Nature – PPSN IX, Lecture Notes in Computer Science, 4193, 182–191, Springer, 2006.

[3] R.K. Ahuja, T.L. Magnanti, and J.B. Orlin: *Network Flows*, Chapter IV of the Handbooks in Operations Research and Management Science, Volume 1: Optimization, (eds. G.L. Nemhauser, A.H.G. Rinnooy Kan, and M.J. Todd), 211–369, North Holland, 1989.

[4] V. Árgilán, J. Balogh, J. Békési, B. Dávid, M. Krész, and A. Tóth: *A flexible system for optimizing public transportation*, In Proceedings of the 8th International Conference on Applied Informatics, Vol. **2**, 181–190, 2010.

[5] V. Árgilán, J. Balogh, J. Békési, B. Dávid, M. Krész, and A. Tóth: *Driver scheduling based on "driver-friendly" vehicle schedules*, In Operations Research Proceedings 2011, Selected Papers of the International Conference on Operations Research (OR 2011), 323–328, Springer, 2012.

[6] V. Árgilán, B. Dávid, Cs. Kemény, G. Pongrácz, and A. Tóth: *Greedy heuristics for driver scheduling and rostering*, In Proceedings of the 2010 Mini-Conference on Applied Theoretical Computer Sciences, Koper, Slovenia, 13-14 October, 2010, University of Primorska Press, 101–108, 2011. (ISBN: 978-961-6832-10-6)

[7] V. Árgilán, J. Balogh, J. Békési, B. Dávid, G. Galambos, M. Krész, and A. Tóth: *An Assignment Model for Real-World Vehicle Scheduling Problem with Refueling*, közlésre benyújtva, 2013.

[8] M. Ball, L. Bodin, and R. Dial: *A matching based heuristic for scheduling mass transit crews and vehicles*, Transportation Science, **7**, 4–31, 1983.

[9] J. Balogh, J. Békési, G. Galambos, and M. Krész: *Model and Algorithm for a Vehicle Scheduling Problem with Refueling*, In Proceedings of the 9th Workshop on Models and Algorithms for Planning and Scheduling Problems, 229–231, 2009.

[10] J. Békési, A. Brodnik, D. Pash, and M. Krész: *An integrated framework for bus logistic management: case studies*, In Logistik Management, Physica-Verlag, 389–411, 2009.

[11] F.C.G. Bellanti, F. Della Croce, and R. Tadei: *A greedy-based neighborhood search approach to a nurse rostering problem*, European Journal of Operational Research, **153(1)**, 28–40, 2004.

[12] A.A. Bertossi, P. Carraresi, and G. Gallo: *On Some Matching Problems Arising in Vehicle Scheduling Models*, Networks, **17**, 271–281, 1987.

[13] L. Bodin, B. Golden, A. Assad, and M. Ball: *Routing and Scheduling of Vehicles and Crews: The State of the Art*, Computers and Operations Research, **10**, 63–211, 1983.

[14] R. Borndörfer: *Discrete Optimization in Public Transportation*, IB-Report 08–56, Konrad-Zuse-Zentrum für Informationstechnik Berlin, Germany, 2008.

[15] R. Borndörfer, A. Löbel, and S. Weider: *A bundle method for integrated multi-depot vehicle and duty scheduling in public transit*, In Computer-aided Systems in Public Transport, (eds. M. Hickman, P. Mirchandani, and S. Voß), Lecture Notes in Economics and Mathematical Systems, 600, 3–24, Springer-Verlag, Heidelberg, 2008.

[16] S. Bunte and N. Kliewer: *An overview on vehicle scheduling models*, Journal of Public Transport, **1(4)**, 299–317, 2009.

[17] P. Cappanera and G. Gallo: *A Multicommodity Flow Approach to the Crew Rostering Problem*, Operations Research, **52(4)**, 583–596, 2004.

[18] A. Caprara, P. Toth, D. Vigo, and M. Fischetti: *Modeling and Solving the Crew Rostering Problem*, Operations Research, **46**, 820–830, 1998.

[19] B.M.W. Cheng, J.H.M. Lee, and J.C.K. Wu: *A nurse rostering system using constraint programming and redundant modeling*, IEEE Transactions in Information Technology in Biomedicine, **1(1)**, 44–54, 1997.

[20] B. Dávid: *Heuristics for the Multiple-Depot Vehicle Scheduling Problem*, In Proceedings of the 2010 Mini-Conference on Applied Theoretical Computer Science, Koper, Slovenia, 13–14 October, 2010, University of Primorska Press, 2011, pp. 23–28. (ISBN: 978-961-6832-10-6)

[21] B. Dávid and M. Krész: *Application Oriented Variable Fixing Methods for the Multiple Depot Vehicle Scheduling Problem*, Acta Cybernetica-Szeged, **21(1)**, 53–73, 2013.

[22] S.W. de Groot and D. Huisman: *Vehicle and crew scheduling: Solving large real-world instances with an integrated approach*, In Computer-aided Systems in Public Transport, (eds. M. Hickman, P. Mirchandani, and S. Voß), Lecture Notes in Economics and Mathematical Systems 600, 43–56, Springer-Verlag, Heidelberg, 2008.

[23] M. Dell' Amico, M. Fischetti, and P. Toth: *Heuristic algorithms for the multiple depot vehicle scheduling problems*, Management Science, **39**, 115–125, 1993.

[24] G. Desaulniers and M.D. Hickman: *Public Transit*, In Handbook in OR & MS, (eds. C. Barnhart and G. Laporte), Vol. **14**, Chapter 2, Elsevier B.V., 2007.

[25] M. Desrochers, J. Gilbert, M. Sauve, and F. Soumis: *CREW-OPT: subproblem modeling in a column generation approach to urban crew scheduling*, In Computer-aided transit scheduling, (eds. M. Desrochers and J.-M. Rousseau), Lecture Notes in Economics and Mathematical Sytems, **386**, 395–406, Springer-Verlag, Berlin, 1992.

[26] M. Desrochers and F. Soumis: *A column generation approach to the urban transit crew scheduling problem*, Transportation Science, **23(1)**, 1–13, 1989.

[27] T.G. Dias, J.P. de Sousa, and J.F. Cunha: *Genetic algorithms for the bus driver scheduling problem: a case study*, Journal of the Operational Research Society, **53(3)**, 324–335, 2002.

[28] A.T. Ernst, H. Jiang, M. Krishnamoorthy, and D. Sier: *Staff scheduling and rostering: A review of applications, methods and models*, European Journal of Operational Research, **153**, 3–27, 2004.

[29] *European Union.* Regulation (EC) No. 561/2006 of the European Parliament and of the Council of 15 March 2006 on the harmonisation of certain social legislation relating to road transport and amending Council Regulations (EEC) 3821/85 and (EC) 2135/98 and repealing Council Regulation (EEC) 3820/85. *Official Journal of the European Union*, L **102**, 11.04.2006.

[30] J.C. Falkner and D.M. Ryan: *EXPRESS: set partitioning for bus crew scheduling in Christchurch*, In Computer-aided transit scheduling, (eds. M. Desrochers and J.-M. Rousseau), Lecture Notes in Economics and Mathematical Systems, 386. 359–378, Springer-Verlag, Berlin, 1992.

[31] R. Freling, C.G.E. Boender, and J.M.P. Paixao: *An integrated approach to vehicle and crew scheduling*, Technical Report 9503/A, Econometric Institute, Erasmus University Rotterdam, Rotterdam, 1995.

[32] R. Freling: *Models and Techniques for Integrating Vehicle and Crew Scheduling*, PhD thesis, Tinbergen Institute, Erasmus University Rotterdam, 1–151, 1997.

[33] R. Freling, A.P.M. Wagelmans, and J.M.P. Paixao: *An overview of models and techniques for integrating vehicle and crew scheduling.* In Computer-Aided Transit Scheduling, (ed. N.H.M. Wilson), Lecture Notes in Economics and Mathematical Systems, **471**, 441–460, Springer, Berlin, 1999.

[34] R. Freling, D. Huisman, and A.P.M. Wagelmans: *Models and algorithms for integration of vehicle and crew scheduling.* Journal of Scheduling, **6**, 63–85, 2003.

[35] A. Gaffi and M. Nonato: *An integrated approach to the extra-urban crew and vehicle scheduling problem*, In Computer-Aided Transit Scheduling, (ed. N.H.M. Wilson), Lecture Notes in Economics and Mathematical Systems, **471**, 103–128, Springer, Berlin, 1999.

[36] M. Gamache, F. Soumis, G. Marquis, and J. Desrosiers: *A column generation approach for large-scale aircrew rostering problems*, Operations Research, **47(2)**, 247–263, 1999.

[37] M.R. Garey and D.S. Johnson: *Computers and Interactability: A Guide to the Theory of NP-Completness*, Freeman, San Fransisco, 1979.

[38] V. Gintner, I. Steinzen, and L. Suhl: *A time-space network based approach for integrated vehicle and crew scheduling in public transport*, In Proc. EWGT2006 Joint Conf., (eds. M. Binetti, F. Civitelle, E. De Liddo, M. Dell'orco, M. Ottomanlli), Bari, Italy, 371–377, 2006.

[39] V. Gintner, N. Kliewer, and L. Suhl: *A Crew Scheduling Approach for Public Transit Enhanced with Aspects from Vehicle Scheduling*, In Computer-aided Systems in Public Transport, (eds. M. Hickman, P. Mirchandani, and S. Voß), Lecture Notes in Economics and Mathematical Systems, **600**, 25–42, Springer-Verlag, Heidelberg, 2008.

[40] N. Guerinik and M.V. Caneghem: *Solving Crew Scheduling Problems by Constraint Programming*, In Proceedings of the 1st Conference of Principles and Practice of Constraint Programming, pp. 481–498, 1995.

[41] K. Haase and C. Friberg: *An exact branch and cut algorithm for the vehicle and crew scheduling problem*, In Computer-Aided Transit Scheduling, Lecture Notes in Economics and Mathematical Systems, **471**, (ed. N.H.M. Wilson), 63–80, Springer, Berlin, 1999.

[42] K. Haase, G. Desaulniers, and J. Desrosiers: *Simultaneous vehicle and crew scheduling in urban mass transit systems*, Transportation Science, **35(3)**, 286–303, 2001.

[43] A. Hadjar, O. Marcotte, and F. Soumis: *A Branch-and-Cut Algorithm for the Multiple Depot Vehicle Scheduling Problem*, Tech. Rept. G–2001–25, Les Cahiers du Gerad, Montreal, 2001.

[44] C. Hane, C. Barnhart, E.L. Johnson, R.E. Marsten, G.L. Nemhauser, and G. Sigismondi: *The fleet assignment problem: Solving a large integer program*, Mathematical Programming, **70(2)**, 211–232, 1995.

[45] D. Huisman: *Integrated and Dynamic Vehicle and Crew Scheduling*, PhD thesis, Erasmus University of Rotterdam, Rotterdam, 2004.

[46] D. Huisman, R. Freling, and A.P.M. Wagelmans: *Multiple-depot integrated vehicle and crew scheduling*, Transportation Science, **39**, 491–502, 2005.

[47] N. Kliewer, T. Mellouli, and L. Suhl: *Solving large multiple-depot multiple-vehicle-type bus scheduling problems in practice*, OR Spectrum, **27(4)**, 507–523, 2005.

[48] N. Kliewer, T. Mellouli, and L. Suhl: *A time-space network based exact optimization model for multi-depot bus scheduling*, European Journal of Operational Research, **175**, 1616–1627, 2006.

[49] A. Kokott and A. Löbel: *Lagrangean Relaxations and Subgradient Methods for Multiplie-Depot Vehicle Scheduling Problems*, ZIB-Report 96–22, Konrad-Zuse-Zentrum für Informationstechnik, Berlin, Germany, 1996.

[50] R.S.K. Kwan, A.S.K. Kwan, and A.S.K. Wren: *Evolutionary Driver Scheduling with Relief Chains*, Evolutionary Computation, **9**, 445–460, 2001.

[51] B. Laurent and J-K. Hao: *Simultaneous Vehicle and Crew Scheduling for Extra Urban Transports*, In New Frontiers in Applied Artificial Intelligence, (eds. N.T. Nguyen, L. Borzemski, A. Grzech, and M. Ali), Lecture Notes in Computer Science Volume, **5027**, 466–475, 2008.

[52] J. Li: *A Self-Adjusting Algorithm for Driver Scheduling*, Journal of Heuristics, **11**, 351–367, 2005.

[53] J-Q. Li and K.L. Head: *Sustainability provisions in the bus-scheduling problem*, Transportation Research, Part D, **49**, 50–60, 2009.

[54] A. Löbel: *Optimal Vehicle Scheduling in Public Transit*, Ph.D. thesis, Technische Universitaet at Berlin, 1997.

[55] A. Löbel: *Vehicle Scheduling in Public Transit and Lagrangian Pricing*. Management Science, **44**, 1637–1649, 1998.

[56] M. Meilton: *Selecting and implementing a computer aided scheduling system for a large bus company*, Algorithms: Combinatorial Analysis. In Computer-Aided Scheduling of Public Transport, (eds. S. Voss and J.R. Daduna), 203–214, Springer-Verlag, Berlin, 2001.

[57] M. Mesquita and J. Paixao: *Multiple Depot Vehicle Scheduling Problem: A New Heuristic Based on Quasi-Assignment Algorithms*, In Computer-Aided Transit Scheduling, (eds. M. Desrochers and J.-M. Rousseau), Lecture Notes in Economics and Mathematical Systems, **386**, 167–180, Springer-Verlag, Berlin, 1992.

[58] M. Mesquita and A. Paias: *Set partitioning/covering-based approaches for the integrated vehicle and crew scheduling problem*, Computers and Operations Research, **35(5)**, 1562–1575, 2008.

[59] M. Moz, A. Respcio, and M. Vaz Pato: *Bi-objective Evolutionary Heuristics for Bus Drivers Rostering*, Working Paper 1–2007, Centro de Investigaao Operacional, Universidade de Lisboa, 2007.

[60] K. Nurmi, J. Kyngas, and G. Post: *Driver Rostering for Bus Transit Companies*, Engineering Letters, **19(2)**, 125–132, 2011.

[61] A.-S. Pepin, G. Desaulniers, A. Hertz, and D. Huisman: *Comparison of Heuristic Approaches for the Multiple Depot Vehicle Scheduling Problem*, Journal of Scheduling, **12(1)**, 17–30, 2009.

[62] A. Rabl: *Environmental benefits of natural gas for buses*, Transportation Research, Part D, **7**, 391–405, 2002.

[63] C.C. Ribeiro and F. Soumis: *A Column Generation Approach to the Multiple-Depot Vehicle Scheduling Problem*, Operations Research, **42(1)**, 41–52, 1994.

[64] J.L. Saha: *An algorithm for bus scheduling problems*, Operational Research Quarterly, **21(4)**, 463–474, 1972.

[65] M. Segal: *The operator-scheduling problem: A network-flow approach*, Operations Research, **24**, 808–823, 1974.

[66] B.M. Smith and A. Wren: *A bus crew scheduling system using a set covering formulation*, Transportation Research, 22A, 97–108, 1988.

[67] I. Steinzen: *Instances for integrated vehicle and crew scheduling problems with multiple depots*, Online elérhető: http://dsor.uni-paderborn.de/index.php?id=bustestset& L=0.

[68] I. Steinzen, V. Gintner, L. Suhl, and N. Kliewer: *A Time-Space Network Approach for the Integrated Vehicle- and Crew-Scheduling Problem with Multiple Depots*, Transportation Science, **4(3)**, 367–382, 2010.

[69] U.U. Suhl, S. Friedrich, and V. Waue: *Progress in solving large scale multi-depot multi-vehicle-type bus scheduling problems with integer programming*, Wirtschaftinformatik Proceedings, Paper **81**, 2007. http://aisel.aisnet.org/wi2007/81

[70] A. Tóth and M. Krész: *A flexible framework for driver scheduling*, In Proceedings of the 11th International Symposium on Operational Research in Slovenia – SOR'11, 341–346, 2011. (ISBN: 978-961-6165-35-8)

[71] A. Tóth and M. Krész: *An efficient solution approach for real-world driver scheduling problems in urban bus transportation*, Central European Journal of Operations Research, **21**(Supplement 1), 75–94, 2013. DOI: 10.1007/s10100-012-0274-3.

[72] A. Wren, S. Fores, A.S.K. Kwan, R.S.K. Kwan, M.E. Parker, and L. Proll: *A flexible system for scheduling drivers*, Journal of Scheduling, **6(5)**, 437–455, 2003.

[73] A. Wren and B.M. Smith: *Experiences with a crew scheduling system based on set covering*, In Computer-aided transit scheduling, (eds. J.R. Daduna and A. Wren), Lecture Notes in Economics and Mathematical Systems, **308**, 104–118, Springer-Verlag, Berlin, 1988.

[74] T. Yunes, A. Moura, and de C. Souza. *Hybrid Column Generation Approaches for Solving Real World Crew Management Problems*, Transportation Science, **39(2)**, 273–288, 2005.

(Beérkezett: 2013. október 4.)

ÁRGILÁN VIKTOR
BALOGH JÁNOS
BÉKÉSI JÓZSEF
DÁVID BALÁZS
GALAMBOS GÁBOR
KRÉSZ MIKLÓS
TÓTH ATTILA
Szegedi Tudományegyetem
Juhász Gyula Pedagógusképző Kar
Informatika Alkalmazásai Tanszék
6701 Szeged, Pf. 396.
{gilan,balogh,bekesi,davidb,galambos,kresz,attila}@jgypk.u-szeged.hu

SCHEDULING PROBLEMS IN THE OPERATIVE PLANNING OF PUBLIC BUS TRANSPORTATION

Viktor Árgilán, János Balogh, József Békési, Balázs Dávid, Gábor Galambos, Miklós Krész, Attila Tóth

Optimization of operative costs is an important aim of public transportation companies. Their planning process can be aided by the optimization of scheduling of vehicle journeys, vehicle schedules, and driver shifts. The problem is a complex optimization problem. For this reason, we deal with the whole problem in three separate phases: vehicle scheduling, driver scheduling and driver rostering. Each sub-problem is NP-hard, and there was no possibility to obtain the optimal solution for large or even middle-sized problems until the last decade. Because of the increase in computational speed and the development of new scheduling methods, researchers today can handle larger problems also. We deal with the mathematical models and solution of these efficient methods. We also refer to our solution methods and the role of the special constraints is analyzed based on our experiences.