

## Előszó

A városi és közúti közlekedési hálózatok vizsgálata és tervezése olyan komplex tudományos-műszaki feladat, amelynek megoldásában nélkülözhetetlen, szabványosított segédeszközzé vált a modern operációkutatási módszerek alkalmazása. Az a (nemzetközi viszonylatban négy-öt, hazánkban három-négy évtizedes múltra visszatekintő) folyamat, amelynek során a matematikai és számítástechnikai modellek és módszerek fokozatosan beépültek a vizsgálatba és tervezésbe, azt is eredményezte, hogy ezek kutatása, fejlesztése, rendszerezése a közlekedéstudomány egy jól behatárolható, viszonylag önálló részterületévé vált, akár tekinthetjük a közlekedési operációkutatás egy ágának is. Ezt igazolja a számos ilyen tárgyú hazai és külföldi tudományos közlemény és konferencia előadás mellett elsősorban az, hogy a témakör a közlekedéstudományi összefoglaló kézi és tankönyvek megfelelő fejezeteiben is megjelent. Hazai példaként hozom az [2], [61] és [36] kézikönyveket, ebből a szempontból külön is kiemelve a [61] megfelelő fejezetét. A szakirodalommal kapcsolatban azonban feltétlenül megemlítendő, hogy újabb keletű, a témakört a kilencvenes évek elejéig-közepéig tárgyaló monográfia csak idegen nyelven hozzáférhető ([33], [62], [65], [69]). A hasonló, magyar nyelvű könyvek hiánya elsősorban a felsőoktatásban érezhető, de nemcsak ott, hiszen egyrészt a modelleket a tervezési gyakorlatban alkalmazó szakemberek is örömmel fogadnának ilyeneket, másrészt időszerű feladat a magyar közlekedéstervezési terminológia aktualizálása és egységesítése is. Mint első kísérletet ezen a téren üdvözölhetjük a [64] távoktatási jegyzetet.

A vizsgálati-tervezési folyamat több fázisra bontható. Először lehatároljuk az érintett területet (röviden *tervezési terület*) felmérjük népességi, terület-felhasználási, közlekedési, szállítási helyzetét. A területet *körzetekre* bontjuk, a valós hálózati elemek bizonyos szintű egyszerűsítésével és absztrakciójával meghatározzuk a *tervezési hálózatot*. Ezután előállítjuk a körzetek közti forgalmat megadó forgalmi mátrixokat: *forgalomkeltés(generation)*, *forgalomszétosztás(distribution)*, *forgalommegosztás(modal split)*. A folyamat ezen pontján bekapcsolódhat ezen mátrixok (számlálási adatok felhasználásával történő) *kiegyenlítése* is. Ha a cél kifejezetten egy távlati tervezés, akkor nem nélkülözhető a hálózati és forgalmi adatoknak a tervezési időszakra történő *előrebecslése(forecasting)*. Ezután következik a *forgalomráterhelés(assignment)*, amelynek során a hálózatban fellépő forgalmi igényeket - valamilyen útvonalválasztási stratégiát feltételezve - elosztjuk (ráterheljük) a hálózat útvonalaira. Egyrészt részleteiben elemezzük a terhelési eredményadatokat, másrészt ezekből további számításokkal, rendezésekkel, csoportosításokkal *komplex* értékelő, minősítő *mutatókat* képezünk, amelyek a *közlekedéspolitikai* döntésekben lényeges szerepet játszanak. Természetesen a gyakorlatban a folyamat iteratív, a végeredmények ismeretében szükségessé válhat egy korábbi fázisra való visszatérés és ismétlés, illetve több variáns párhuzamos vizsgálata.

Megjegyzem, hogy a hazai ilyen irányú kutató-fejlesztő munkák és alkalmazások első csúcsidezőszaka a hetvenes évek végén, nyolcvanas évek elején a *budapesti metróépítés* újraindításához kapcsolódik. A kilencvenes években új lendületet adott ennek a területnek is az *autópályahálózat* kibővítésének tervezése (különös tekintettel az útdíjas autópályákra).

Dolgozatom fő témája a *forgalomráterhelési* fázis, operációkutatási szempontból, különös tekintettel az ebben kulcsfontosságú szerepet játszó *útvonalválasztási* eljárásokra, valamint bizonyos, a gyakorlatból adódó *speciális tervezési helyzetek* modellezhetőségére.

---

*Célom a meglevő eljárások, algoritmusok továbbfejlesztése, valamint új megoldások kidolgozása volt.*

Az alábbiakban összefoglalom a dolgozat tartalmát:

- Az 1. fejezetben részletesen tárgyalom a ráterhelési eljárást, az elméleti háttér, a pontos feladat-meghatározás és a dolgozat további részében használatos terminológia tisztázása céljából. Áttekintem a megoldási módszereket. Itt írom le azt az általános ráterhelési sémát, amely a kapacitáskorlátos forgalomráterhelés egy megvalósítása, és amelynek egyes moduljaival a későbbiekben részletesen foglalkozom, illetve amelybe a későbbiekben beillesztem a speciális modellezési megoldásokat.
- A 2. fejezet témája a tervezési hálózat matematikai-számítástechnikai modellezése. Itt definiálom az alaphálózatból a modellhálózatot létrehozó hálózati transzformációkat, amelyek feladata az, hogy egyrészt a matematikai algoritmusokkal kezelhető, másrészt a valós tervezési helyzet és a speciális igények modellezésére alkalmas hálózatot hozzanak létre. A közlekedési hálózat mint matematikai hálózat is speciális, vannak olyan jellemzők amelyek elkülönítik általában a hálózatokon belül, illetve más speciális hálózatoktól. Az algoritmusok hatékonyságának javítása szükségessé teszi a közlekedési hálózat fogalom minél pontosabb, kvantitatív megfogalmazását, valamint a specialitásokat minél hatékonyabban kihasználó számítógépes tárolási adatstruktúrák definiálását. Itt - a ráterhelés szempontjából tekintve - összehasonlító elemzésnek vetem alá az általában alkalmazott adatstruktúrákat, és definiálok egy olyan variánst, amelyre alapozok a dolgozat további részeiben.
- A 3. fejezetben foglalkozom a ráterhelési feladat útvonalválasztási moduljával. Bár az útvonalválasztás az alkalmazásoknál kiterjedhet több szempontra és több útvonalra is, a számítástechnikai realizáció központi, és a végrehajtási idő szempontjából meghatározó része az az algoritmus, amely itt már egy absztrahált, matematikai hálózaton az élhosszak alapján előállítja a legrövidebb (első minimális) utakat. Áttekintem és elemzem az erre a célra általánosan a legjobbnak tartott fa-építő eljárásokat, definiálok egy új algoritmust, igazolom helyességét és hatékonyságát, számítógépes futási eredményekkel is dokumentálom hasznosságát. Ugyancsak megadok egy hatékony, a faépítésen alapuló algoritmust is, a második minimális utak meghatározására.
- A 4. fejezetben foglalkozom a ráterhelés specialitásaival. Az alkalmazások igényei az általános modellen belül számos problémát, speciális esetet vetnek fel. A dolgozatban az alábbiakat tárgyalom, ismertetve az általam kidolgozott és alkalmazott megoldási javaslatokat is:
  - A forgalom kitiltása egyes hálózatrészekről.
  - Egyes hálózatrészekre vett relatív forgalmak előállítása.
  - Az autópályadíjak figyelembevétele, többfajta útdíjképzési és kedvezményezési módszer esetére.
  - Részhálózatok beágyazott forgalmi vizsgálata.
- Az 5. fejezetben egy konkrét megvalósítást írok le. A dolgozatban tárgyalt elvek és módszerek alkalmazásaként valósult meg egy több évig fejlesztett, általam tervezett, és csoportmunkában kivitelezett, nagy méretű és komplexitású számítógépes programrendszer (NETWINFO), amely - gyakorlati tervezési feladatokhoz - széleskörű hazai alkalmazásra került és néhány külföldi felhasználási referenciával is rendelkezik, valamint a Széchenyi István Főiskolán szakirányú oktatási anyagként is szolgál. Ennek felépítését, főbb paramétereit és szolgáltatásait ismerteti ez a fejezet.

- A 6. fejezetben áttekintem az általam ismert, vagy legalább ismertető szinten elérhetővé vált, a magyar oktatási és/vagy tervezési gyakorlatban a jelen időszakban (a kilencvenes években) használatos ráterhelési programcsomagokat.
- A *Függelék* gráfelméleti alapfogalmak és számítógépes tesztelési eredmények mellett a NETWINFO programrendszer főbb alkalmazási referenciáit tartalmazza.

Ezúton is köszönetet mondok *Dr. Bakó András* egyetemi tanárnak, aki a hetvenes évek végén, nyolcvanas évek elején a győri Közlekedési és Távközlési Műszaki Főiskolán a Számítástechnika tanszék és a Számítóközpont vezetőjeként eltöltött néhány éve alatt elindította a főiskolán "*Számítástechnikai módszerek alkalmazása a közlekedés tervezésében és irányításában*" témájú kutatási-fejlesztési főirányt és bevezetett erre a területre. A vele azóta is fennálló szakmai kapcsolatnak jelentős szerepe van abban, hogy a dolgozat elkészült.

Köszönöm *Dr. Koren Csaba* egyetemi tanárnak azt, hogy a *közlekedéstervezési* tudásbázis, a magyar és idegen nyelvű közlekedéstudományi szakirodalom megismerésében segített.

Köszönetemet fejezem ki *dr. Kálmán László* főiskolai adjunktusnak, akivel a közvetlen gyakorlatban, konkrét és valós közlekedéstervezési feladatok megoldásában dolgoztunk sokat együtt. Ennek a közös munkának jelentős szerepe van abban, hogy az említett programrendszer első verziói óta a felhasználás igényeihez igazodva sokat fejlődött, szolgáltatási szintje lényegesen javult.

A NETWINFO programrendszer elkészítésében - a grafikus szolgáltatások programjainak kidolgozásával - jelentős szerepe van *Pusztai Pál* főiskolai adjunktusnak. A korábbi verziók programozási munkáiban részt vett még *Orbán Gábor* programtervező matematikus és *Prédl Antal* közlekedésmérnök.

Külön köszönetet mondok *Kövesné Dr. Gilicze Éva* egyetemi tanárnak, a BME Közlekedésmérnöki Kar Közlekedésüzemi Tanszék vezetőjének, hogy témaválasztásomat támogatta és a dolgozat első fogalmazványát részletes és érdemi bírálatnak vetette alá. Az ő és munkatársai által megfogalmazott észrevételek figyelembevételével, a jelzett hibák és hiányosságok korrekciója megítélésem szerint mind formai, mind tartalmi szempontból lényegesen jobbá tette dolgozatomat.

## 1. A forgalomráterhelési feladat

### 1.1. A feladat megfogalmazása

A vizsgálati-tervezési folyamat egyik fő komponense a *ráterhelési eljárás*, amelynek során a hálózatban fellépő forgalmi igényeket - valamilyen útvonalválasztási stratégiát feltételezve - elosztjuk (ráterheljük) a hálózat útvonalaira. Az eljárás indulásakor - megelőző tervezési fázisok eredményeképpen - már adott a hálózat, a körzetbeosztás, a körzetek közt fellépő forgalmi igények. Az eljárás eredménye az egyes hálózati elemek (szakaszok, csomópontok) forgalma, *forgalmi terhelése*, amely további - összegző, értékelő - számítások kiindulásaként szolgál.

A ráterhelési eljárások közös elvi alapmodellje az *egyensúlyi* (equilibrium) modell, ez két alapvető feltevésre támaszkodik, amelyek közül egyik az utazóra, másik a hálózatra vonatkozik:

- Az *utazó*, hogy célját elérje, a lehetséges útvonalak közül a minimális költségű útvonalat választja.
- Az *útvonalak* és ezen belül a hálózati elemek fajlagos igénybevételi költsége növekszik, ha a forgalmi terhelés nő.

Ezeket komplex formában fejezi ki a jól ismert Wardrop féle első alapelv [72]:

- *Egy kiindulási és célpont közt igénybevett útvonalak utazási költsége kisebb vagy egyenlő mint az ugyanezen pontok közt lehetséges, de utazásra fel nem használt útvonalak utazási költsége.*

Az utazási költségen itt is természetesen a fajlagos, az egy utazásra jutó költség értendő. Mint látjható, az utazó viselkedése, útvonalválasztási stratégiája az egyensúlyi modellben tudatos, a hálózat és a forgalmi helyzet ismeretén alapul, tehát a modell alkalmazásánál feltételezzük, hogy az utazások döntő többsége ilyen, a hely és a helyzet előzetes ismeretével lebonyolított utazás.

Az egyes hálózati elemek költségfüggvényei (általánosabb terminológiával: az *ellenállásfüggvények*) tehát a forgalom monoton növekvő (pontosabban nemcsökkenő) függvényei. A forgalomnövekedésnek van egy természetes korlátja, amely az elérhető sebességnek és az illető hálózati elem forgalomátbocsátó kapacitásának korlátozottságából következik. Az ellenállásfüggvény az alábbi általános formában adható meg [33]:

Az  $F$  forgalomnagysághoz tartozó fajlagos utazási költség  $K(F)$  :

$$K(F) = K(0) * \left( 1 + \alpha * \left( \frac{F}{C} \right)^\beta \right), \text{ ahol}$$

- $K(0)$  a nulla forgalomnagysághoz (üres hálózat) tartozó fajlagos utazási költség  
 $F$  az aktuális forgalomnagyság  
 $C$  a forgalmi kapacitás  
 $\alpha, \beta$  nemnegatív paraméterek

A forgalomnagyság és kapacitás adatok szokásos mértékegysége az *egységjármű/óra*. Részletesebb függvényeket és elemzéseket a saját programcsomaggal kapcsolatban fogok adni.

A függvény matematikailag egy konvex függvény. Ebből következően az egyensúlyi követelmény megfogalmazható, és a feladat megoldható matematikai programozással, pontosabban egy *konvex* célfüggvényes és lineáris feltételrendszerrel rendelkező *minimalizálási* feladatnak egy iterációs eljárásban való ismételt alkalmazásával is. Az ilyen jellegű megközelítések jó összefoglalását és egy-egy megvalósítását találhatjuk a [21] és [3] publikációkban.

Az *egzakt optimalizálási* eljárások matematikai szempontból nézve elegánsak és elvben a kiinduló hipotézisnek megfelelő megoldást szolgáltatnak. Ennek ellenére a gyakorlatban nem nevezhető egyeduralkodónak ez a modellcsoport. Egyrészt nagyobb méretű gyakorlati feladatoknál a feltételrendszer változóinak száma igen nagy, a feladat számítástechnikailag nehezen kezelhető, másrészt a tervező által a megoldási folyamat nem eléggé átlátható.

A mai közlekedéstudományi alap és tankönyvek (pl.[33], [62], [69]) is standard eszközként tárgyalják és a közlekedéstervezési gyakorlatban ma is széleskörűen használják a *heurisztikus* megoldási technikákat, ezen belül - a fenti alakú ellenállásfüggvényekkel dolgozó - *többlepcsős* kapacitáskorlátos *ráterhelést*. Az ilyen eljárások előnyös tulajdonságaként hozható fel:

- *Számítástechnikai* szempontból:
  - A tárigény és a futási idő jól becsülhető, nagyméretű feladatok is megoldhatók általánosan hozzáférhető számítástechnikai eszköztárral, elfogadható időn belül.
- *Tervezői* szempontból:
  - A megoldási folyamat átlátható, egyes paraméterei ill. fázisai közvetlenül kapcsolatba hozhatók a közlekedési fogalmakkal, részfolyamatokkal.
  - A tervező tapasztalatai, mérnöki intuíciói könnyen átvihetők a modellbe, az egyes módosítások és következményeik közti kapcsolat tervezői szemmel nézve is követhető, érthető.
  - Módot ad olyan jellegű eredményadatok képzésére is, amelyek csak a ráterhelés, a forgalom-felépülés *folyamatában* gyűjthetők, magából a végeredményből már nem állíthatók vissza.
  - Az eljárás, megfelelően paraméterezve, a tervezési gyakorlatban jól használható, az optimumhoz közelálló megoldást ad.

### 1.2. A lépcsős ráterhelés

Dolgozatom anyagának egy jelentős része (a hálózat és az útvonalválasztás modellezése) alapjában véve független az ráterhelési módszertől, illetve bármely módszer részeként alkalmazható. A dolgozat más részei viszont erősen kötődnek egy jól körül határolható ráterhelési eljárástípushoz (amelynek alkalmazásával a saját fejlesztésű programcsomagunk is készült). Ez a már fentebb említett *lépcsős (kapacitáskorlátos) ráterhelés*, angol terminológiával: *incremental assignment (with capacity-restraint)*, a német szakirodalomban: *Sukzessivumlegung*. A dolgozatban betöltött kiemelt szerepe és a terminológia rögzítése céljából a megfelelő *alapeljárást* itt előrebocsátva vázolom.

A ráterhelési eljárás indulásakor már - korábbi tervezési fázisok eredményeképpen - rendelkezésre állnak az alábbi adatok:

- A tervezési-vizsgálati *alaphálózat* (a továbbiakban röviden: alaphálózat), amely *csomópontokból* és *élekből* (a szakaszok egyes irányából) áll, és a valódi hálózathoz képest az aktuális cél szempontjából megengedhető és célszerű *egyszerűsítésekkel*, *általánosításokkal* jön létre. A csomópontokat és éleket a továbbiakban röviden *hálózati elemeknek* nevezzük. Nem (vagy csak összevontan) szerepelnek a kevésbé lényeges elemek, és a hálózatba bevont elemek fizikai és közlekedési jellemzőit is néhány globális paraméterbe tömörítjük. Ezek az egyes elemekre vonatkozóan elsősorban:
  - forgalomszabályozási adatok
  - forgalmi kapacitás adatok
  - hossz adat (éleknél)
  - egyéb, az ellenállásfüggvény paramétereibe beépülő adatok
- A tervezési-vizsgálati *forgalmi körzetek* (a továbbiakban röviden: *körzetek*). Egy körzet egy összefüggő területrésznek felel meg a hálózatban, mint csomópontok egy halmaza is tekinthető.
- A tervezési-vizsgálati *időszak* (a továbbiakban röviden: *időszak*), az időintervallum, amelyre a forgalmi alapadatok, a forgalmi mátrixok vonatkoznak, pl. egy óra vagy egy nap.
- A körzetszintű *forgalmi mátrixok* (a továbbiakban röviden: *körzetmátrixok*), amelyek az elosztandó forgalmat tartalmazzák a kiinduló körzet-célkörzet (más elnevezéssel *forrásnyelő* körzet) viszonylatokban. Ezek a mátrixok *forgalmi rétegenként* adottak. Egy forgalmi rétegen (a továbbiakban röviden: *réteg*) értem a forgalom egy viszonylag homogén, egységes módon paraméterezhető szeletét (például tehergépjárművek, személygépkocsik és/vagy átmenő forgalom, belső forgalom stb.). A körzetmátrixokban az adatok általában természetes egységekben - mint pl. az adott időszakra vonatkozó járműdarabszám - vannak megadva. Ezt a mértékegységet a továbbiakban röviden *mátrixegységnek* nevezzük.
- *Rétegjellemzők*. Egy-egy forgalmi réteghez hozzárendelhetők olyan adatok, amelyek a réteget jellemzik és a rétegspecifikus *ellenállásfüggvénybe* épülnek be. Ilyenek lehetnek például:
  - fajlagos (1 utazás 1 kilométerére jutó) üzemeltetési költség
  - az idő értéke (1 utazás 1 órájára jutó, általános értelemben vett utazási költség)
  - egyéb tényezők pl. egységjármű/óra mértékegységbe átváltó szám

Magához a ráterhelési eljáráshoz meg kell adni:

- A konkrét *ellenállásfüggvényeket*. Ezek elvben rétegenként és ezen belül hálózati elemenként eltérőek lehetnek, de a gyakorlatban - a modellezendő valósággal összhangban - a hálózati elemekből csoportokat, *kategóriákat* képezünk, és egy kategórián belül azonos módon paraméterezünk (útkategóriák pl. 6 sávós autópálya-szakasz, csomóponti kategóriák pl. lámpás csomópont). Így végül is az ellenállás számítási módja kategóriától és rétegtől függ. Az ellenállásfüggvények a forgalmat általában standardizáltan, egységjármű/óra mértékegységben kezelik.
- A ráterhelési *lépcsőzés* adatait. Egy lépcső a ráterhelendő forgalom egy része, egy körzetmátrix (egy réteg) egy bizonyos százaléka. A *lépcsőzés* a lépcsőknek és ráterhelési sorrendjüknek a definiálása. A lépcsőzésen belül a rétegek keveredhetnek is (egymás utáni lépcsők lehetnek különböző rétegekből). A lépcsőzés megválasztása tervezői feladat, mérnöki megfontolásoknak, tapasztalatoknak a konkrét helyzetre való alkalmazása, bár vannak erre vonatkozó általános szakirodalmi ajánlások is [61].

- A körzetenkénti *forrás/nyelő* (a továbbiakban röviden: *f/ny*) pontokat és ezeknek a forgalom kibocsátásában ill. elnyelésében való relatív *súlytényezőit*. Ezeket a súlytényezőket más megfogalmazásban tekinthetjük a körzetbeli *f/ny* pontokban való utazás kezdési/befejeződési *valószínűségeloszlás* leíró adatainak is. Az *f/ny* pontok és súlyok helyes meghatározása a tervezési folyamat egy eléggé nehéz, a hálózat és a forgalmi viszonyok beható ismeretét megkövetelő része, mindazonáltal nem megkerülhető, mivel útvonalválasztást kell modellezni, egzaktan meghatározható útvonalak viszont csak pontok (és nem pl. körzetek) között léteznek. Ez a probléma nem a lépcsős ráterhelés specifikuma, minden olyan ráterhelési módszernél fellép, ahol útvonalakkal dolgozunk. Az *f/ny* pont és súlyrendszer általában *rétegfüggő*, tehát *rétegenként* kell megadni. Ehhez szorosan kötődik a körzetmátrixoknak az *f/ny* pont és *súly* rendszernek megfelelő felbontása, a *forrás-nyelő körzet* forgalom *forráspont-nyelőpont* forgalmakká való átalakítása. Az így létrejött mátrixokat *pontszintű forgalmi mátrixoknak*, röviden *pontmátrixoknak* nevezzük.



1. ábra. Lépcsős ráterhelés

Ezek előrebocsátása után az 1. ábrán foglalom össze a lépcsős ráterhelési eljárás általános algoritmusát. Ehhez megjegyzem, hogy a *terhelések inicializálása* azt jelenti, hogy az első lépcső előtt alapesetben a hálózat üres, vagyis minden terhelési érték nulla, de speciális tervezési célokhoz dolgozhatunk *előtöltéssel* is, amikor is egyes hálózati elemek valamilyen kezdeti terheléssel indulnak.

A lépcsős ráterhelési eljárás által szolgáltatott eredmények elvi értelemben vett helyessége (az elvi egyensúlyi állapot megközelítési foka) az ellenállásfüggvények és a rétegek megválasztásán túlmenően azon múlik, hogy mi a lépcsők sorrendje és milyen részletes a lépcsőzés. Minél kisebbek az egyes lépcsők, annál hatékonyabban működnek az ellenállásfüggvények, annál egyenletesebb lesz a forgalmi terhelés a hálózaton.

### 1.3. Más heurisztikus eljárások

A lépcsős ráterhelés mellett a gyakorlatban alkalmazott programcsomagok más heurisztikákat is használnak az elvi egyensúlyi állapot közelítő meghatározására. Ezek számítástechnikai modellezése ugyan nem képezi dolgozatom tárgyát, mindazonáltal a terminológiai tisztázás céljából és abból az okból, hogy ezeknek a *számítástechnikai modellezés* szintjén nagyon sok közös eleme van a lépcsős ráterheléssel, a szakirodalom (elsősorban [62]) alapján röviden összefoglalom ezek főbb típusait.

#### 1.3.1. Mindent vagy semmit eljárás

A *mindent-vagy-semmit* (all-or-nothing) típusú forgalomráterhelés tekinthető a lépcsős ráterhelés azon speciális esetének is, amikor a teljes ráterhelendő forgalmat *egy rétegbe* fogjuk össze, és az *egyetlen lépcső* ennek 100 % - a. Ekkor a kapacitáskorlát nem érvényesülhet, az esetleges irreális túlterheléseket (és emellett viszonylag sok hálózati elem terheletlenül maradását) az ellenállásfüggvény nem tudja megakadályozni. Ez a módszer önmagában csak nagyon speciális esetekben alkalmazható, inkább más eljárások részeként alkalmazott.

#### 1.3.2. Iterációs eljárások

Ezek a *mindent-vagy-semmit* eljárás bizonyos *konvergenciakritériumok* és/vagy *lépésszámhatárok* által vezérelt ill. korlátozott számú ismételtetésén, iterációján alapulnak. Két alapvető *kritérium* van egyik az *ellenállások* a másik a *forgalmi terhelések* stabilitása.

##### *Ellenállás-iteráció*

1. Inicializáljuk a hálózati elemeket nulla forgalommal és meghatározzuk az ehhez tartozó ellenállásértékeket.
2. Útvonalválasztás az aktuális ellenállásértékek mellett és a *teljes* forgalom ráterhelése.
3. Határozzuk meg a hálózati elemeknek a ráterhelt forgalom melletti ellenállásértékeit. (Egy variáns: ezekből és a korábbi lépésekben kialakult értékekből képezzünk valamilyen súlyozott átlagot, és ezt tekintjük a jelen iterációs lépéshez tartozó ellenállásként.)
4. Ha az ellenállások eltérése az előző lépés ellenállásértékeitől egy előre megadott hibahatáron belül van, vagy elértük a lépésszámkorlátot, akkor készen vagyunk. Ellenkező esetben töröljük a forgalmat a hálózatról és ismételjünk a 2. ponttól.

##### *Forgalom-iteráció*

1. Inicializáljuk a hálózati elemeket nulla forgalommal és meghatározzuk az ehhez tartozó ellenállásértékeket.
2. Útvonalválasztás az aktuális ellenállásértékek mellett és a teljes forgalom ráterhelése.
3. Határozzuk meg a hálózati elemek új terhelését, mint a megelőző iterációs lépésben kialakult terhelés és a 2. pontban adódott terhelés valamilyen súlyozott átlagát.



4. Ha a terhelések eltérése az előző iterációs lépés terhelésértékeitől egy előre megadott hibahatáron belül van, vagy elértük a lépésszámkorlátot, akkor készen vagyunk. Ellenkező esetben számoljuk újra az ellenállásértékeket a jelen iterációs lépés terheléseivel és ismételjünk a 2. ponttól.

Természetesen a módszerek illetve a konvergenciakritériumok kombinációjával számos iterációs eljárásvariáns létrejöhet. Az eljárásokban említett átlagszámításokhoz használt súlyozó tényezők a helytől és a tervezői megfontolásoktól függő empirikus értékek.

### 1.4. Egzakt optimalizálási eljárások

*Egzakt* optimalizálási eljárásoknak nevezzük azokat a megoldási módszereket, amelyek valamely - matematikai értelemben véve - pontosan definiált algoritmust hajtanak végre és az algoritmus által megszabott - szintén matematikai jellegű - feltételek teljesülése esetén a követelményeknek megfelelő megoldást adnak. Dolgozatomnak nem célja ezek tárgyalása, itt csak terminológiai okokból, a későbbi hivatkozások kedvéért említem meg ezek fő típusait:

- *Determinisztikus egyéni optimum (deterministic user equilibrium DUE)*  
Mint a fejezet bevezetőjében már említettem, az egyensúlyi követelmény megfogalmazható, és a feladat megoldható matematikai programozással, pontosabban egy *konvex* célfüggvényes és lineáris feltételrendszerrel rendelkező *minimalizálási* feladatnak egy iterációs eljárásban való ismételt alkalmazásával is. Minden iterációs lépésben végrehajtódik egy mindent-vagy- semmit típusú ráterhelés és egy lineáris optimalizáció. Az eljárás folyamán a hálózati elemek terhelése konvergál az egyensúlyi követelményt kielégítő megoldáshoz. A megoldási módszer alapvetően a kvadratikus programozási feladat egy speciális esetéből származik [22], több implementációja is ismert és alkalmazott [27], [40].
- *Sztohasztikus egyéni optimum (stochastic user equilibrium SUE)*  
Ez a determinisztikus eljárás általánosításaként is tekinthető, olyan értelemben, hogy itt nem tételezzük fel az utazóról a körülmények teljes ismeretét, az utazó viselkedése, útvonalválasztási stratégiája bizonytalansági, valószínűségi elemeket is tartalmaz. Ennek következtében az ilyen modellek többé-kevésbé szimulációs jelleggel is bírnak. Az általános bizonytalansági elvnek természetesen több konkrét megvalósítása létezik [62], [67].
- *Rendszeroptimum (system optimum SO)*  
Ebben az esetben az optimalizáció célja a hálózat teljes utazási költségének minimalizálása. Az optimumhoz képest bármely utazás áthelyezése csak növelheti a teljes költséget, akkor is, ha ez a konkrét utazás számára kedvezőbb lenne (ez lehetséges, mivel a rendszeroptimum nem feltétlenül az egyéni optimumok összessége). A feladat matematikai modellje és megoldási módszere a determinisztikus egyéni optimuméhoz hasonló, annál némileg egyszerűbb. Mivel a kiinduló hipotézis gyakorlati esetekben nem reális, a modell csak nagyon speciális esetekben alkalmazott.

### 1.5. A tömegközlekedési feladat

Bár még kifejezetten nem utaltam rá, de az eddigiekből is következik, hogy dolgozatom tárgya elsődlegesen az általános, vagy más szóval *egyéni közlekedési* optimalizálási modell, amely alapvetően a *járművek* közlekedésével foglalkozik, és egyik fő jellemzője ezek *szabad* útvonalválasztása.

A lehetséges másik nagy témakör, amit itt csak a terminológiai teljesség kedvéért említünk meg, az egyén mozgása, közlekedése a tömegközlekedési hálózatban - beleértve a szükségszerűen beiktatandó gyalogos szakaszokat is - vagyis a *tömegközlekedési* tervezési, optimalizálási feladat. Ez számos speciális modellezési és adatkezelési problémát vet fel, ilyenek például:

- a tömegközlekedési vonalak és a járatok, a járatgyakoriságok,
- a közös szakaszokon közlekedő különböző járatok,
- az átszállások, várakozások

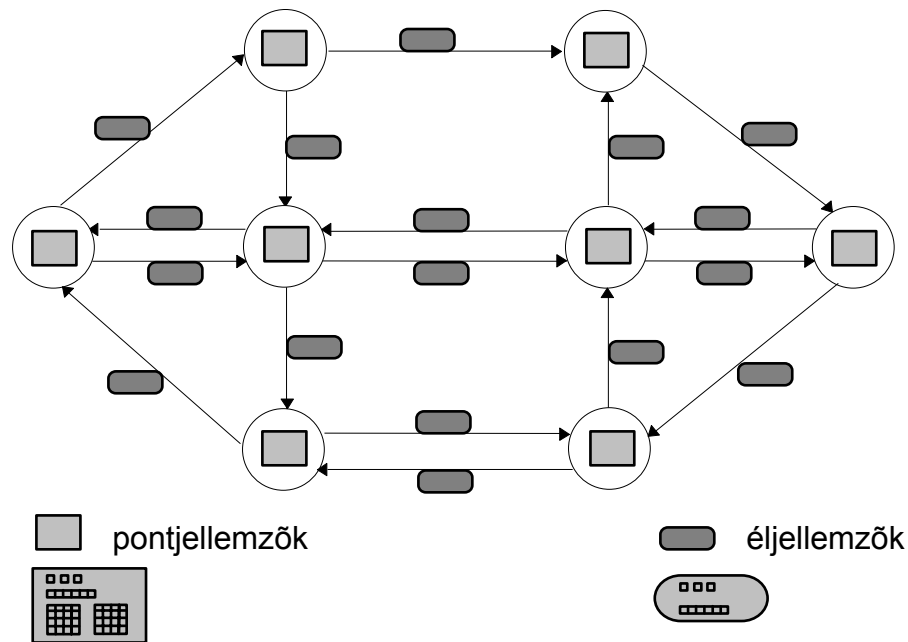
megadása illetve modellezése.

Bár a kétféle feladat nagyon sok közös elemet tartalmaz (ilyenek például a legrövidebb utakat kereső algoritmusok) valamint lehetséges, és egyszerűbb modellekben elegendő is az egyéni közlekedési modellre való visszavezetés (például a szakaszok összefonódásait feloldhatjuk fiktív csomópontok beiktatásával, az átszállásokat kezelhetjük csomóponti ellenállási problémaként), az eltérések, specialitások indokolják, hogy a nagyobb, igényesebb és a teljességre törekvő tervező rendszerek erre a feladatra külön modellt alkalmazzanak.

## 2. A hálózati modell

### 2.1. Alapfogalmak

Elsősorban a terminológia és jelölésrendszer tisztázása céljából előrebocsátva foglalkoznom kell néhány a dolgozat további részében használandó gráf és hálózatelméleti alapfogalommal. Ezek jól ismert matematikai alapdefiníciói a függelékben találhatóak (1. pont). A közlekedési hálózatnak a ráterhelési és hasonló jellegű feladatoknál alkalmazott modellje ezekkel az elnevezésekkel az alábbiakban foglalható össze:



2. ábra. Alaphálózati modell

#### Hálózati modell

- *Hálózati gráf:* Egyszerű, irányított és összefüggő gráf. A pontok az úthálózati csomópontoknak, az élek a csomópontokat közvetlenül összekötő szakaszok haladási irányainak felelnek meg. Szakasonként egy (egyirányú szakasz) vagy kettő él lehet a modellben. Ez a leképezés az alapesetre érvényes, a hálózat transzformációinál más értelmet is nyerhet.
- *Pontjellemzők:* Pontonként a ponthoz rendelt értékek. Ezek lehetnek egyszerű értékek (pl. koordináták), de bonyolult adatstruktúrák is (pl. a forgalomirányítást leíró adathalmaz).
- *Éljellemzők:* Élenként az élhez rendelt értékek. Ezek általában egyszerű értékek (pl. hossz, sebesség, idő).

Konkrét és részletesen kidolgozott példát a pont és éljellemzőkre az 5. fejezetben adok.

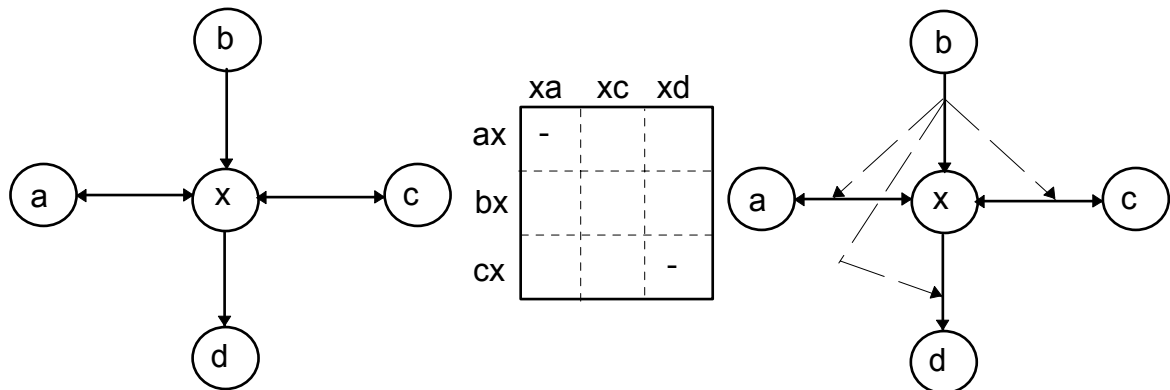
### 2.2. Hálózat-transzformációk

A tervezési hálózat közvetlen leképezése egy irányított hálózattá általában nem elegendő ahhoz, hogy egyrészt a *matematikai algoritmusokkal kezelhető*, másrészt a *valós tervezési helyzet* és *igények* modellezésére alkalmas hálózatot hozzon létre.

Erre a forgalomráterhelési eljárásokban legkézenfekvőbb példa a csomópontok kezelése. Általában (és nem feltétlenül csak a városi hálózatokban) a csomópontoknak is van nem elhanyagolható ellenállása, és a csomóponton belüli forgalmak is érdekesek lehetnek az eredményadatok között. A csomóponti probléma modellezésének általános elvi módszere régóta ismert [34], mindazonáltal az egy-egy konkrét modellben való megvalósítása valamint az ebből adódó, ehhez csatlakozó modellezési megoldások egyediek lehetnek [48].

### 2.2.1. A csomóponti transzformáció

Egy útvonal megtétele során *szakaszokon* és *csomópontokon* haladunk át, tehát az útvonal teljes ellenállása az útba eső élellenállások és *élkapcsolat-ellenállások* (csomóponti ellenállások) összege. Ha például minimális ellenállású utakat keresünk, ezekkel kell számolni. A *útkeresés* számítástechnikai algoritmusai jól ismertek, a hálózat *méretét*, a hálózat *strukturális* tulajdonságait, valamint a rendelkezésünkre álló számítástechnikai *programfejlesztő* környezetet és *programfuttató*, alkalmazó számítástechnikai bázist figyelembe véve, mindig optimálisan választhatunk tár és számítási idő szempontjából *hatékony*, elméletileg *garantált helyességű* útkereső (első, második ... stb. legkedvezőbb út meghatározó) eljárást.



3.ábra. Csomópont felbontása

Viszont ezeknek az algoritmusoknak közös jellemzője az, hogy egy, a gyakorlat szempontjából nézve *egyszerűsített* hálózatot tételeznek fel, *ellenállása csak az éleknek lehet*, a pontokhoz ellenállásadat nem rendelhető. A probléma egy lehetséges (és valószínűleg legjobb) feloldása az, hogy a hálózati modellt igazítjuk az algoritmusok kívánalmaihoz, vagyis egy alkalmasan megválasztott előzetes csomóponti *transzformációt* hajtunk végre. Ennek alapelve:

- *A transzformált hálózatban az alaphálózat éleiből pontok, élkapcsolataiból pedig élek lesznek.*

Tehát felbontjuk a csomópontot a benne meglévő *bemenő-kimenő* élkapcsolatok összességére. A csomópont kiinduló és eredményadatai így *élkapcsolati* típusú mátrixokkal írhatók le (3. ábra). A mátrix egy eleme a *sorél-oszlopél* mint *bemenőél-kimenőél* kapcsolathoz rendelt érték. Az ábrán az áttekinthetőség kedvéért csak a **(b, x)** él mint bemenő él kapcsolatait tüntettük fel, ezek mind egyirányúak, hiszen az **x** pontban a **b** pont felé nem lehet haladni a szakasz egyirányúsítása miatt.



A példából látható, de általánosan is érvényes, hogy a transzformáció lényegesen (2-4 -szeresére) megnöveli a pontok és élek számát, tehát a hálózat méreteit. Ennek ellenére alkalmazása elkerülhetetlen, ha kellő pontossággal és részletességgel kívánunk modellezni. A transzformáció *közlekedéstervezői* szempontból nézve is *szemléletes* és jelentést hordozó, hiszen teljesen elemeire bontja a hálózatot.

### 2.2.2. A forrás - nyelő pontok beillesztése

Külön megoldandó részfeladat a *forgalomindítás* és *elnyelés transzformálása*. Ha az alaphálózatban a forgalom pontokból indul (*forráspontok*), illetve pontokban ér célhoz (*nyelőpontok*), akkor a transzformált hálózatban a megfelelő elemek *élek* lennének. Viszont egy konkrét élhez rendelés korlátozná az útvonalak választékát, mivel például a forgalom indításánál (az alaphálózatra visszavetítve) a forráspontból kiinduló *egyik meghatározott* éllel indulna *minden* útvonal, kizárva a többi élt.

A probléma fiktív pontok és élek felvételével kezelhető. A példán (4a. ábra) legyen az **a** és a **h** forrás/nyelőpont. Az alaphálózatban mindegyikhez felveszünk egy *fiktív*, csak erre a célra szolgáló, és csak ezekhez a pontokhoz csatlakozó segédpontot, az **a** -hoz az **A** -t és a **h** -hoz a **H** -t. A forgalom a modell alaphálózatban az ilyen pontokból indul és ezekbe érkezik (a csatoló élek ellenállása természetesen nulla), tehát ezek veszik át a tényleges f/ny pontok szerepét. Mivel ezeknek csak egyetlen (kétirányú) kapcsolata van, a transzformáció egyértelmű és nem korlátozza az útvonalválasztást. A transzformált hálózatban az alaphálózatbeli f/ny pont kettéválik a fiktív csatoló él két irányának (indítás, elnyelés) megfelelően az **(A, a)** és **(H, h)** típusú élekből lesznek a forráspontok, az **(a, A)** és **(h, H)** típusúakból pedig a nyelőpontok.

### 2.2.3. Az eredmények leképezése

A ráterhelési eljárás magja, az útkeresés-útválasztás és az utakra való ráterhelés a transzformált hálózatban hajtódik végre. Viszont az eredményeket a felhasználó nyilván az *alaphálózatban* akarja (esetleg pl. grafikus megjelenítésben) *szemlélni, értékelni*. Tehát szükség van (a konkrét ráterhelési eljárástól függően az eljárás folyamán akár többször is) az eredményadatok alaphálózatra való *leképezésére*, visszatranszformálására. A leképezés végrehajthatóságához természetesen szükség van az alap és transzformált elemek kapcsolatát leíró információkra, ezeket meg kell őriznünk, célszerű módon tárolnunk kell őket. Ezzel részleteiben a következő fejezetben foglalkozom, itt most tételezzük fel, hogy ezek az információk rendelkezésünkre állnak.

A ráterhelési eljárásban a legfőbb eredmények a hálózati elemek *terhelésadatai*, de sok esetben használható és esetleg más szempontból továbbfeldolgozható eredményadatok az útvonalkeresés közvetlen eredményei maguk a forrás/nyelő pontok közti *utak* is. Az alábbiakban ezért az ilyen utak és a terhelések leképezésével foglalkozom.

Az *utak leképezése* a transzformációs információk birtokában eléggé egyszerű:

- *A transzformált hálózatbeli út alaphálózati megfelelőjét úgy kapjuk, hogy a (transzformált) pontsor minden eleméhez a megfelelő alaphálózatbeli él kezdőpontját rendeljük.*
- *Az útvonal összellenállását a transzformált hálózatbeli út éellenállásainak összegzésével kapjuk.*

Vegyük példaként a 4a. ábrán vastag vonallal bejelölt transzformált és alaphálózatbeli utat. Az

**Aa - ac - cd - dg - gf - fh - hH**

út megfelelője az

**a - c - d - g - f - h**

út, az út tényleges ellenállása 122.

Megjegyzendő, hogy a transzformált hálózatbeli *körmentes* útból az alaphálózatban lehet *körös* is. Például a következő összetett útkeresési feladat:

- az **A** pontból indulva először a legkedvezőbb úton jussunk el az **f** -be, ezután
- az **f** -ből a legkedvezőbb úton az **e** -be, végül
- az **e** -ből a legkedvezőbb úton a **H** pontba.

Mivel az **f** - pontban a **c** -ből jövet nem lehet balra fordulni, és a modell nem tartalmazza a csomópontban való megfordulás (pontosabban: visszafordulás) lehetőségét sem, a megoldás a transzformált hálózatban az

**Aa - ac - cf - fh - hg - gf - fe - eh -hH**

útvonal, ami az alaphálózatra visszavetítve az

**a - c - f - h - g - f - e - h**

útvonal.

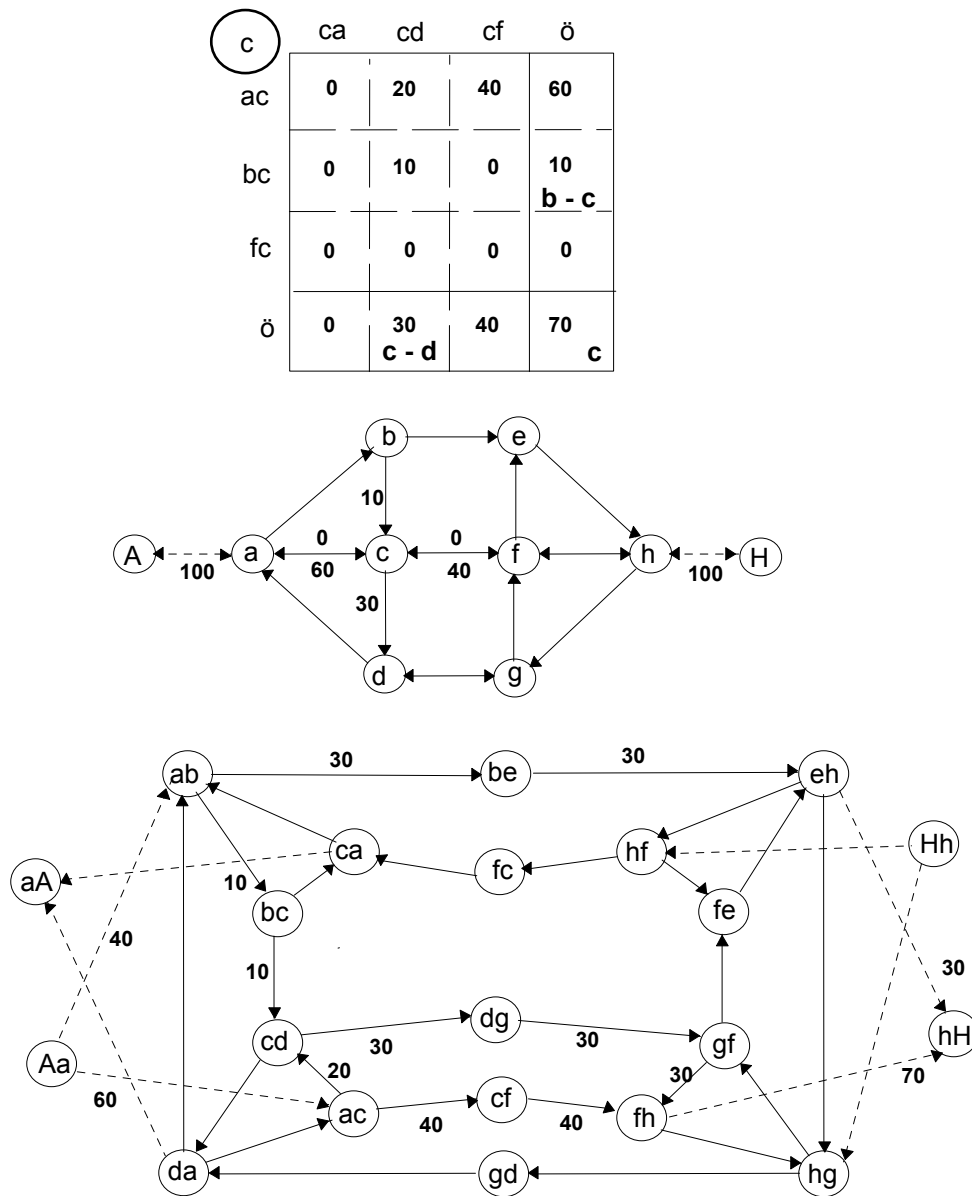
A *terhelési eredményeket* illetően az alaphálózati él és pontterhelések szükségesek. Erre vonatkozóan:

*Az alaphálózatbeli él terhelését azon transzformált élek terheléseinek összegeként kapjuk, amelyek végpontja az alaphálózatbeli élnek megfelelő transzformált pont.*

A szabály következményeként további hasznos (esetlegesen ellenőrzési célra is felhasználható) információkat is kaphatunk a fiktív alaphálózati élekre vonatkozóan:

- *Az alaphálózatbeli fiktív f/ny pontba befutó él terhelése a valós f/ny pont teljes elnyelt forgalma.*
- *Az alaphálózatbeli fiktív f/ny pontból kiinduló él terhelése nulla.*
- *A transzformált hálózatbeli forráspontból kiinduló élek terheléseinek összege a megfelelő alaphálózati f/ny pont teljes kibocsátott forgalma.*

A terhelési eredmények visszaállítását a 4b. ábrán szemléltetem. Az ábrán az élekre és a mátrixba írt számok itt terheléseket jelentenek. Az alaphálózati éleknél az egyértelműség kedvéért a "menetirány szerinti" jobb oldalon vannak a számok. Csak a példaként hozott terhelési adatokat tartalmazza az ábra. A csomóponti élkapcsolati mátrix ki van bővítve összegekkel.



4b. ábra. Csomóponti transzformáció: terhelések

A 4b. ábra hálózatán:

- a **(c, d)** él terhelése az **(ac, cd)** és a **(bc, cd)** élek terheléseinek összege
- a **(h, H)** él terhelése az **(eh, hH)** és az **(fh, hH)** élek terheléseinek összege, a **h** pont teljes elnyelt forgalma.
- az **(Aa, ac)** és az **(Aa, ab)** élek terheléseinek összege az **a** pont teljes kibocsátott forgalma.

Az alaphálózati *pontterheléseket* élkapcsolatonkénti bontásban közvetlenül a transzformált hálózatbeli élterhelések adják, ezeket beírva az alaphálózati élkapcsolati mátrix megfelelő elemébe. (A kihúzott helyekre természetesen nullát írunk.) Azokat az éleket vesszük figyelembe, amelyek a kérdéses pontbani élváltást jelentenek az alaphálózatban, vagyis ha **x** jelöli a kérdéses pontot, minden **(\*x, x\*)** alakú transzformált hálózatbeli él terhelése az **x** pont terhelésének egy összetevőjét adja. A mátrix *összege* a pont *összterhelése*. A mátrix sorösszegei a megfelelő *befutó*, oszlopösszegei a megfelelő *kiinduló* alaphálózati élek teljes



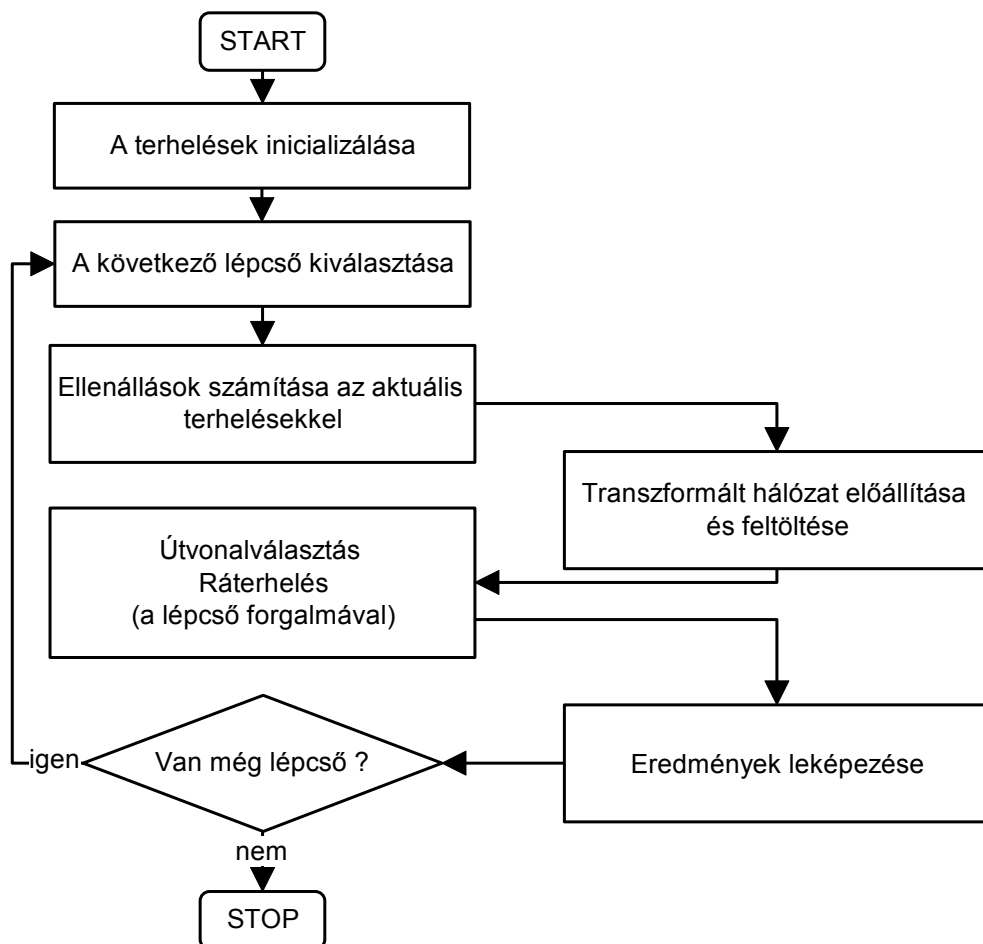
terhelését adják. Könnyen látható, hogy a pontterhelési mátrixokból minden (pont, él, élváltás) terhelési adat kinyerhető.

A 4b. ábrán nézzük a  $c$  alaphálózati pont példáját. A pont élkapcsolati mátrixába a transzformált hálózat  $(ac, cd)$ ,  $(ac, cf)$ ,  $(bc, ca)$ ,  $(bc, cd)$  és  $(fc, ca)$  éleinek forgalmát írjuk be a megfelelő helyekre. A második sor összege a az alaphálózati  $(b, c)$  él forgalma, a második oszlop összege az alaphálózati  $(c, d)$  él forgalma.

#### 2.2.4. Beillesztés a ráterhelési eljárásba

Lépcsős ráterhelési eljárásomban a *transzformációt* és az *eredmények leképezését lépcsőnként* végre kell hajtani. Ennek két fő oka:

- a *forgalomszabályozás* rétegfüggő lehet (pl. teherautóval behajtani tilos)
- a *forrás/nyelő* pontok rendszere és súlyozása is rétegenként változhat.



5. ábra. Transzformáció a ráterhelésben

Ezért tehát miután kiszámítottuk az aktuális ellenállásokat az alaphálózaton, ezek és az *aktuális réteg* ismeretében előállítjuk a transzformált hálózatot, majd feltöltjük az aktuális terhelésekkel az aktuális élkapcsolati mátrixokból.

Az útvonalválasztások és ráterhelések után le kell képeznünk az aktuális eredményeket az alaphálózatra, mivel a következő lépcsőben a transzformált hálózat egészen más is lehet mint az előző.

Megjegyzem, hogy a transzformált hálózat rétegenkénti megőrzése - ami az ismételt transzformációkat elkerülhetővé tenné - nem szükséges, mert a transzformáció és a leképezés egy gyors, szekvenciális folyamat és mivel az alaphálózatot és az aktuális transzformált hálózatot az operatív tárban tudjuk tartani (ld. következő pont), ezen műveletek időigénye - a szükségképpen lépcsőnként végrehajtandó és az időigényben domináns - útvonalválasztáshoz képest elenyésző. A módosított folyamatot az 5. ábrán adom meg.

### **2.3. Adatok és tárolásuk**

Egy számítógépes operációkutatási modellben mindig központi feladat a kezelendő adatok helyes tárolási struktúrájának megválasztása. Ez erősen összefügg az alkalmazható-alkalmazandó algoritmusok körével és hatékonyságával. Általában az *adatstruktúra-algoritmus* függés, egymásra hatás kölcsönös, a hatékony modell egy többlépéses iterációs módosítási, egymáshoz igazítási folyamat eredménye. Az előzőekben - legalábbis minőségileg - tisztáztuk, hogy a forgalomráterhelési feladat milyen adatokat kezel. Az is látható, hogy az adatok nagyobb része a hálózati struktúrához, a hálózati elemekhez kötődik. Előrebocsátom (és ez a dolgozat további részeiben egyértelműen igazolódni fog), hogy a ráterhelési eljárás hatékonysága szempontjából a hálózati elemekhez közvetlenül nem köthető adatok (pl. rétegjellemzők, forgalmi mátrixok) tárolási struktúrája csekély jelentőséggel bír az előbbiekhöz képest. Ennek következtében *a fő feladat a hálózati szerkezet helyes leképezése a számítástechnikai adatstruktúrák szintjére.*

#### **2.3.1. A hálózati adatstruktúra**

A modellben két fajta hálózat szerepel, az *alaphálózat* és a *transzformált hálózat*. Az alaphálózat magán viseli a közúti közlekedési hálózatok közvetlenül tapasztalható jellemzőit: *Az egy ponthoz csatlakozó élek száma univerzálisan egy viszonylag kis konstans értékkel korlátozható, és átlagos értéke (a gráfelméleti maximumhoz képest) nagyon alacsony.* Az utóbbi tulajdonsággal rendelkező hálózatokat "*ritka*" hálózatnak nevezzük. A transzformáció ezen a tulajdonságon lényegesen nem változtat, a hálózat ezután is ritka marad, vagyis az átlagos élszám kicsi, de a csatlakozó élek száma egyes pontoknál és egyes speciális transzformációknál (ld. a dolgozat későbbi részében) nagymértékben meg is nőhet. (A hálózatok ritkaságának mértékére vonatkozó konkrét számadatokat az útvonalválasztási eljárásokkal kapcsolatos példaanyagban láthatunk.)

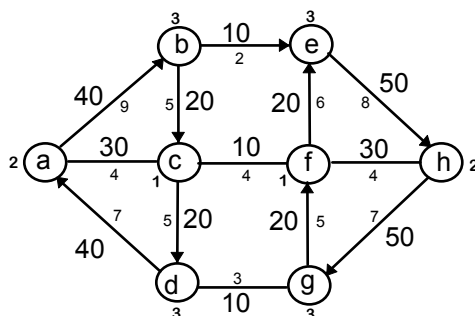
Az adatszerkezetek megválasztásánál és minősítésénél általában három fő tényező veendő tekintetbe:

- a *tárkihasználás* és ezzel összefüggően az operatív tárban való *tárolhatóság*
- a *karbantarthatóság* vagyis a változások átvezetésének műveletigénye
- a *lekérdezhetőség* vagyis az információkinyerés műveletigénye.

Az *első tényező* számításiigényes feladatoknál - tehát itt is - *kiemelt fontosságú* (különös tekintettel az eljárás magját, és a számításiigény szempontjából meghatározó részét alkotó *útvonalválasztás és ráterhelés* részmodulra) hiszen a jelenlegi számítástechnikai eszközbázisra nézve általánosan igaz, hogy a háttértár használata több nagyságrenddel lelassítja a számításokat.

A ráterhelés szempontjából a *második* tényező elhanyagolható, mivel itt nincs a hagyományos értelemben vett karbantartási feladat. Mind az aktuális ellenállások számítása, mind hálózattranszformáció, mind az eredmények leképezése egy egy menetes (iterációmentes) szekvenciálisan végrehajtható folyamat, bármely ésszerű adatstruktúrán a szükséges minimális lépésszámmal végrehajtható.

A *harmadik* tényezőt illetően itt jellemző lekérdezési mód az útkereső, útvonalválasztó algoritmusok futtatása, ezeken belül a pont és éladatok kötetlen sorrendű (random) elérése. A szóhajóhető adatszerkezeteket e szempontok szerint fogom vizsgálni.



6. ábra. Tárolási példa

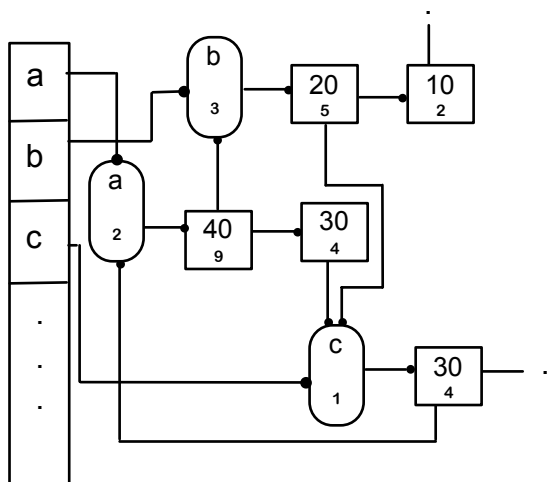
A tárolási módszerek illusztrálására a 6. ábra példahálózatát használom. Az ábrán pontonként egy (a pont mellé írt kis szám) pontjellemző és élenként kettő (az él mellé írt kis és nagy szám) éljellemző van, az utóbbiak a kétirányú élek két irányánál azonosak. A következő idevonatkozó ábrákon a pontjellemzőt **T**, az éljellemzőket **H** illetve **K** jelöli. Ezek konkrét jelentése itt közömbös.

A hálózat tárolására egy közvetlenül adódó egyszerű lehetőség a *statikus, mátrixos* tárolás (7. ábra). A pontjellemzők (a pontsorszámmal indexelt) egy, az éljellemzők kétdimenziós tömbök elemei. A *lekérdezhetőség* nagyon jó, az adatokat a lehető leggyorsabban, indexeléssel érjük el. Nagyon hatékony útkereső algoritmusokkal lehet kezelni ezt az adatszerkezetet. Sajnos, gyakorlati méretű feladatoknál mégsem választható a mátrixtárolás a hatalmas tárigény miatt. A közlekedési hálózatok, még tervezési szinten is általában több ezer pontot tartalmaznak, éljellemzőnként (bájtban mérve) tízmillió nagyságrendű a tárigény, ami nem is csökkenthető, mert az erre az adatszerkezetre alapozott algoritmusok a kezdetben ritka mátrixot teljesen feltöltik. Keresni kell a tárolási redundancia csökkentését, esetleg a másik két szempont rovására is.

		<b>T</b>		<b>H</b>	1	2	3	4	5	6	7	8		<b>K</b>	1	2	3	4	5	6	7	8
<b>a</b>	<b>1</b>	2		<b>1</b>	-	40	30	-	-	-	-	-		<b>1</b>	-	9	4	-	-	-	-	-
<b>b</b>	<b>2</b>	3		<b>2</b>	-	-	20	-	10	-	-	-		<b>2</b>	-	-	5	-	2	-	-	-
<b>c</b>	<b>3</b>	1		<b>3</b>	30	-	-	20	-	10	-	-		<b>3</b>	4	-	-	5	-	4	-	-
<b>d</b>	<b>4</b>	3		<b>4</b>	40	-	-	-	-	-	10	-		<b>4</b>	7	-	-	-	-	-	3	-
<b>e</b>	<b>5</b>	3		<b>5</b>	-	-	-	-	-	-	-	50		<b>5</b>	-	-	-	-	-	-	-	8
<b>f</b>	<b>6</b>	1		<b>6</b>	-	-	10	-	20	-	-	30		<b>6</b>	-	-	4	-	6	-	-	4
<b>g</b>	<b>7</b>	3		<b>7</b>	-	-	-	10	-	20	-	-		<b>7</b>	-	-	-	3	-	5	-	-
<b>h</b>	<b>8</b>	2		<b>8</b>	-	-	-	-	-	30	50	-		<b>8</b>	-	-	-	-	-	4	7	-

7. ábra. Mátrixtárolás

A memóriaigényt csökkentendő, gondolhatunk egy *dinamikus* összetett *listaszerkezetre* is, amely kétféle, egy pont és egy éljellegű listaelemből (rekordból) építkezik, a jellemzőket a rekordban tárolva, a kapcsolatokat mutatókkal (pointer) megvalósítva (8. ábra). Ennek tárigénye kedvező, hiszen az adatokon kívül csak a pointereknek kell tárhely. Az elérés már kevésbé jó, hiszen listákon való lépkedéssel jutunk el az éljellemzőkhöz. A szerkezet dinamikus jellegét igazán nem tudjuk hasznosítani (a karbantartás nem a fő szempont).



8. ábra. Dinamikus adatszerkezet

Az általam a feladathoz legjobbnak tartott ábrázolási mód *statikus, egydimenziós tömböket* használ, mind a pont mind az éljellemzők tárolására. Az adatszerkezetet a 9. ábra szemlélteti. Mint ebből látható, az éljellemzőket tömören, egy a kezdőpont szerint rendezett sorban tároljuk, az **M** mutatóérték mondja meg, hogy a sorban mely szakasz tartozik egy-egy kezdőponthoz, és ez a **V** (végpontindex) értékkel együtt határozza meg az élt. Az **i** és a **j** egyszerű sorszámok, indexek. Az adatszerkezet *tárkihasználása* nagyon jó, és mivel a közlekedési hálózatoknál az egy pontból kiinduló élek száma a hálózat méretétől függetlenül korlátozott, a *memóriaigény* a hálózat *méretével* csak *lineárisan* (és nem négyzetesen mint a mátrixnál) nő, több ezer pontos hálózat is tárolható a ma már elérhető, átlagosnak tekinthető professzionális személyi számítógépes memóriaméretben (16 - 64 Mbájt). Az adatelérés is jónak mondható, mivel egy ponthoz egy, egy élkezdőponthoz kettő (**M** mutató) indexezéssel elérhetünk, innen az éladatot már ugyan keresnünk kell, de ez (az egy pontból kiinduló élek számának korlátozottsága miatt) csak néhány (átlagosan 3 körüli) lépést igényel a tömbben.

	a	b	c	d	e	f	g	h
<b>i</b>	1	2	3	4	5	6	7	8
<b>T</b>	2	3	1	3	3	1	3	2
<b>M</b>	1	3	5	8	10	11	14	16

<b>j</b>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
<b>V</b>	2	3	3	5	1	4	6	1	7	8	3	5	8	3	6	6	7
<b>H</b>	40	30	20	10	30	20	10	40	10	50	10	20	30	10	20	30	50
<b>K</b>	9	4	5	2	4	5	4	7	3	8	4	6	4	3	5	4	7

9. ábra. Éltárolás

Levonható tehát a következtetés, hogy a fenti - rövid elnevezéssel - *éltárolási* módszer a *közlekedési hálózatokhoz* és a *ráterhelési feladathoz* jól illeszkedő közel optimális

*adatstruktúra*, mivel helykihasználása jó, a szokásos méreteknél az operatív tárban elhelyezhető, a pont és éljellemezőkhöz való hozzáférés gyors. Maga a tárolás a ritka mátrixok számítógépes tárolási elvi módszerének egy konkrét alkalmazása [4].

Ezután vizsgáljuk meg, hogy milyen jellegű adatokat tárolunk ebben a struktúrában a modellünkben szereplő két fajta hálózatban. Hangsúlyozom, hogy itt egy elvi adatmodellről van szó, amely majd a később ismertetendő ráterhelési programcsomagnál kap konkrét, részletesen kifejtett aktuális tartalmat.

### **2.3.2. Alaphálózati adatok**

#### **2.3.2.1. Pontjellemezők**

- *Kiinduló adatok*
  - *Csomóponttípus*: a csomópont kiépítésére és a csomóponti forgalomszabályozás módjára jellemző adat, tulajdonképpen azt a függvényt (és annak egyes paramétereit) határozza meg, amely szerint irányonként a csomóponti ellenállásértékeket számítjuk.
  - *Pont-körzetjellemző*: a csomópont körzethez rendelését, és a pontnak a körzetben játszott szerepét - *forrás/nyelő* pont jelleg, *súlytényező* értéke - határozza meg. A körzetmátrix-pontmátrix átalakításnál és a hálózat-transzformációnál (f/ny pontok beillesztése) használjuk fel.
  - *Áthaladásszabályozás*: egy *élkapcsolati típusú mátrix*, amely az adott csomópontban érvényes speciális forgalomszabályozási helyzetet írja le (pl. balrakanyarodás tiltása bizonyos irányra). A csomóponti transzformációnál használjuk fel.
- *Eredményadatok*
  - *Csomóponti forgalom*: egy *élkapcsolati típusú mátrix*, amely az adott csomópontban a ráterhelés eredményeképpen adódó forgalmi terhelési értékeket tartalmazza. Az "eredmények leképezése" lépésben jön létre. Rétegenként és összesen szerepel.
  - *Egyéb eredményadatok*: a ráterhelés folyamatában számítható és szükség szerint tárolható *élkapcsolati típusú mátrixok* pl. csomóponti ellenállás( idő, költség), késleltetés, idővesztés. Rétegenként és összesen szerepelhetnek.
- *Technikai adatok*: esetleges kiegészítő funkciók céljából szükség lehet további adatokra, pl. *grafikus* kezeléshez, szemléltetéshez pontkoordinátákra.

Megjegyzem, hogy az eredményadatok *rétegenkénti* is számítása és tárolása inkább csak információs célú, a ráterhelés menetében a következő ráterhelési lépések útvonalválasztásához szükséges ellenállás-számítások az *összforgalmi adatokkal* történnek.

Megjegyzendő még, hogy az *élkapcsolati típusú mátrixok* (az egy ponthoz csatlakozó élek számának korlátozottsága miatt) korlátozottak és kis méretűek, tárolási szempontból egyszerű pontjellemezőként kezelhetők.

#### **2.3.2.2. Éljellemzők**

- *Kiinduló adatok*
  - *Szakasztípus*: az útszakasz forgalomtechnikai kiépítését, fizikai-műszaki jellemzőit és a forgalom lebonyolódását befolyásoló egyéb tényezőket (pl. sebességkorlát) leíró adat, az ellenállásfüggvény típusát és egyes paramétereit határozza meg.
  - *Szakaszhossz*: az ellenállásfüggvény paramétereként szolgál.

- *Szakasz-rétegjellemzők*: rétegenként a szakaszra adott speciális jellemzőket (ezek általában forgalomszabályozási jellemzők, mint pl. behajtási tilalom, sebességkorlát) adják meg. Ezeket a csomóponti transzformációnál és az ellenállás számításánál használjuk fel.
- *Eredményadatok*
  - *Élforgalom*: Az adott élen a ráterhelés eredményeképpen adódó forgalmi terhelési értéket tartalmazza. Az "eredmények leképezése" lépésben jön létre. Rétegenként és összesen szerepel.
  - *Egyéb eredményadatok*: a ráterhelés folyamatában számítható és szükség szerint tárolható *éladatok* mint pl. ellenállás ( idő, költség), sebesség, késleltetés, idővesztés, kapacitáskihasználtság. Rétegenként és összesen szerepelhetnek.
- *Technikai adatok*: esetleges kiegészítő funkciók céljából szükség lehet további adatokra, pl. szemléltetéshez út(utca) nevekre.

Megjegyzem, hogy *rétegenkénti* eredményadatokra a pontjellemzőknél mondtak itt is érvényesek.

### **2.3.3. Transzformált hálózati adatok**

Mint az eddigiekből következik, a transzformált hálózat az a tiszta matematikai hálózat, amelyet az útvonalválasztás helyes számíthatósága céljából hozunk létre - a forgalmi helyzet rétegfüggősége miatt - minden ráterhelési lépcsőben. Ennek a hálózatnak csak *éljellemzői* vannak. Ezek:

- *Ellenállásérték*: Az él aktuális forgalmi ellenállása.
- *Élforgalom*: Az élen a ráterhelés eredményeképpen adódó aktuális teljes forgalmi terhelési érték.
- *Lépcső-élforgalom*: Az élen a ráterhelés aktuális lépcsőjéből származó forgalmi terhelési érték. Számítása csak az alaphálózati élforgalom rétegenkénti gyűjtéséhez szükséges.

### 3. Útvonalválasztás a hálózaton

#### 3.1. Bevezetés, jelölések

A fejezet a témája a ráterhelési feladat *útvonalválasztási* modulja. Bár az útvonalválasztás az alkalmazásoknál kiterjedhet több szempontra és több útvonalra is, a számítástechnikai realizáció központi, és a végrehajtási idő szempontjából meghatározó része az az algoritmus, amely itt már egy absztrahált, matematikai hálózaton (tehát a transzformált hálózaton) az élhosszak alapján előállítja a legrövidebb (minimális hosszú) utakat. A minimális utak meghatározásának feladata általában - konkrét alkalmazásoktól függetlenül - is a hálózatok tanulmányozásának egyik alapvető és sokszor érintett témaköre, úgy az általános hálózati, mint a közlekedési hálózati tématerületen számos publikáció tárgya. A témakör összefoglaló jellegű tárgyalását és bő bibliográfiáját találhatjuk az általános hálózatokra vonatkozóan a [17], míg a közlekedési hálózatokra vonatkozóan a [24] cikkben.

Az itt ismertetendő algoritmusok tömör és pontos leírásának céljából bevezetek illetve itt is rögzíték néhány jelölést:

- A hálózat pontjait betűkkel jelölöm :  $a, b \dots$
- A hálózat pontjainak számát jelölje :  $N$ .
- Az  $a$  pontból a  $b$  pontba mutató él jelölése:  $(a, b)$  .
- Az  $(a, b)$  él hosszát jelölje:  $h(a, b)$  .
- A tömb indexhivatkozások jelölésére a szögletes zárójelet használom, pl.  $T[x, y]$  .
- A halmazok elemfelsorolásos jelölésére a kapcsos zárójelet használom, pl:  $\{a, b\}$ ,  $\{a\}$ ,  $\{\}$
- A halmazok tartalmazási viszonyainak jelölésére a  $<, >, =$  jeleket használom, ahol a  $<$  jel a részhalmaz viszonyt jelenti, a halmazműveleteket a  $+$  és  $-$  jelöli.
- Egy  $x$  kiinduló és  $y$  célpontból álló pontpárt egy  $x-y$  *viszonylatnak* nevezek és jelölök.

#### 3.2. Minimális utak

##### 3.2.1. Mátrix eljárás

Az alább ismertetendő eljárást a szakirodalom - a meghatározó alapelv első publikálójáról - Warshall féle eljárásnak nevezi [73]. Akkor alkalmazzuk, ha minden viszonylatban (vagy legalábbis a viszonylatok nagy részében) meg akarjuk határozni a minimális utat, és van elegendő operatív tárterületünk az algoritmus által igényelt két darab  $N \times N$  méretű mátrix tárolására.

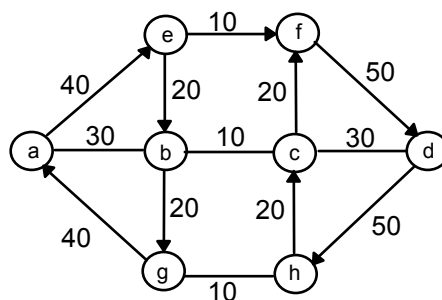
Az algoritmus könnyebb leírásához vezessünk be egy újabb fogalmat: Az, hogy egy  $x-y$  *viszonylat* minimális útját egy  $w$  pont *bevonásával* keressük, azt jelenti, hogy az utat egy  $x-w$  kezdő és egy  $w-y$  befejező részútból próbáljuk összerakni (vagyis az  $x$  -ből az  $y$  -ba a  $w$  - n keresztül megyünk).

Az algoritmus egy  $T$  távolságmátrixot és egy  $C$  cíkmátrixot használ. Mindkettő  $N \times N$  méretű és soronként és oszloponként is a pontokkal (vagy a pontsorszámokkal) van indexelve (úgy mint a hálózat mátrixos tárolásánál). A sorindex egy viszonylat kezdőpontjának, az oszlopindex egy viszonylat végpontjának felel meg.

A  $\mathbf{T}$  elemei a viszonylatok *aktuális távolságát* vagyis az aktuális minimális útjának *hosszát* tartalmazzák, tehát a  $\mathbf{T}[\mathbf{x}, \mathbf{y}]$  az  $\mathbf{x}$ - $\mathbf{y}$  viszonylat ilyen távolsága. A  $\mathbf{C}$  elemei *pontok* (vagy pontindexek) és a minimális út összerakásához, tehát a megfelelő pontsorozat előállításához szükséges adatokat tartalmazzák abban a formában, hogy a  $\mathbf{C}[\mathbf{x}, \mathbf{y}]$  az a pont, amely az  $\mathbf{x}$ - $\mathbf{y}$  viszonylat aktuális minimális útján az  $\mathbf{x}$  kezdőpont után jön (merre induljunk a kezdőpontból a végpont felé).

Az algoritmus a  $\mathbf{T}$  és a  $\mathbf{C}$  egy kezdőállapotából kiindulva, a mátrixokat *lépésenként javítva*, több lépés megtétele után jut el a végeredményhez. Mint látható, pontosan annyi lépés kell, ahány pont van a hálózatban. Ezek után az algoritmus a következőképpen definiálható:

- *Kezdőállapot*  
 $\mathbf{T}$  : ha a kezdő és végpont között van közvetlen összeköttetés (él), akkor ennek hossza a távolság, egyébként a távolság végtelen nagy. (Mint látható, ez tulajdonképpen a hálózat mátrixos tárolásának megfelelő mátrix.)  $\mathbf{C}$  : a viszonylat címke eleme a végpont, annak megfelelően, hogy a  $\mathbf{T}$  kezdőállapota tulajdonképpen egy élből álló utakat jelent.
- *Javító lépések*  
 A hálózat minden  $\mathbf{w}$  pontjára (egyszer) és ezen belül minden  $\mathbf{x}$ - $\mathbf{y}$  viszonylatra (egyszer) végrehajtandó: kíséreljük meg a  $\mathbf{w}$  -t *bevonni* a viszonylatba, vagyis ha  $\mathbf{T}[\mathbf{x}, \mathbf{y}] < \mathbf{T}[\mathbf{x}, \mathbf{w}] + \mathbf{T}[\mathbf{w}, \mathbf{y}]$  (vagyis a  $\mathbf{w}$  bevonásával rövidítünk), akkor legyen az új távolság  $\mathbf{T}[\mathbf{x}, \mathbf{y}] = \mathbf{T}[\mathbf{x}, \mathbf{w}] + \mathbf{T}[\mathbf{w}, \mathbf{y}]$  és az új címke  $\mathbf{C}[\mathbf{x}, \mathbf{y}] = \mathbf{C}[\mathbf{x}, \mathbf{w}]$  (vagyis az  $\mathbf{y}$  -hoz vezető minimális úton ugyanúgy kell indulni, mint a  $\mathbf{w}$  -hez vezető minimális úton).
- *Végállapot*  
 A  $\mathbf{T}$  végállapota megadja a *tényleges minimális út* hosszát egy-egy viszonylathoz. Ez az érték már nem csökkenthető, tehát ennél rövidebb út nem található. Ha van olyan pont a hálózatban, amelyből vagy amelybe egyáltalán nincs út, tehát a pont valamelyik, vagy mindkét értelemben *elszigetelt* (izolált), akkor a megfelelő mátrixelemek végtelen értékűek maradnak. Maguk az *útvonalak* kiolvashatók a  $\mathbf{C}$  -ből, a címkemátrix definíciója alapján.



10. ábra. Mátrixmódszer példa

Az algoritmushoz a 10.ábra hálózatán adok példaadatokat. A 11a. ábra a kezdőállapotot, a 11b. ábra az első javítás ( $\mathbf{w}=\mathbf{a}$ ), a 11c. ábra a második javítás ( $\mathbf{w}=\mathbf{b}$ ) utáni állapotot, a 11d. ábra a végállapot. Például határozzuk meg a példahálózatban az  $\mathbf{f}$ - $\mathbf{g}$  viszonylatra az utat, a végállapot ábra segítségével  $\mathbf{C}[\mathbf{f}, \mathbf{g}] = \mathbf{d}$ ,  $\mathbf{C}[\mathbf{d}, \mathbf{g}] = \mathbf{c}$ ,  $\mathbf{C}[\mathbf{c}, \mathbf{g}] = \mathbf{b}$ ,  $\mathbf{C}[\mathbf{b}, \mathbf{g}] = \mathbf{g}$ , tehát az út:  $\mathbf{f}-\mathbf{d}-\mathbf{c}-\mathbf{b}-\mathbf{g}$  a hossza  $\mathbf{T}[\mathbf{f}, \mathbf{g}] = 110$ .

A javító lépések algoritmususa tehát *három egymásba ágyazott* ciklus. A külső a bevonandó pont szerinti, ezen belül van a viszonylat kezdőpontja szerinti, és a legbelső a viszonylat



végpontja szerinti ciklus. Az elvi algoritmus sorrendet nem ír elő, a példában - mint legegyszerűbbet - a *növekvő* sorrendet alkalmazzuk mind a három ciklusban.

T	a	b	c	d	e	f	g	h	C	a	b	c	d	e	f	g	h
a	0	30	~	~	40	~	~	~	a	a	b	c	d	e	f	g	h
b	30	0	10	~	~	~	20	~	b	a	b	c	d	e	f	g	h
c	~	10	0	30	~	20	~	~	c	a	b	c	d	e	f	g	h
d	~	~	30	0	~	~	~	50	d	a	b	c	d	e	f	g	h
e	~	20	~	~	0	10	~	~	e	a	b	c	d	e	f	g	h
f	~	~	~	50	~	0	~	~	f	a	b	c	d	e	f	g	h
g	40	~	~	~	~	~	0	10	g	a	b	c	d	e	f	g	h
h	~	~	20	~	~	~	10	0	h	a	b	c	d	e	f	g	h

11a. ábra. Mátrixmódszer kiindulás

T	a	b	c	d	e	f	g	h	C	a	b	c	d	e	f	g	h
a	0	30	~	~	40	~	~	~	a	a	b	c	d	e	f	g	h
b	30	0	10	~	70	~	20	~	b	a	b	c	d	e	f	g	h
c	~	10	0	30	~	20	~	~	c	a	b	c	d	e	f	g	h
d	~	~	30	0	~	~	~	50	d	a	b	c	d	e	f	g	h
e	~	20	~	~	0	10	~	~	e	a	b	c	d	e	f	g	h
f	~	~	~	50	~	0	~	~	f	a	b	c	d	e	f	g	h
g	40	70	~	~	80	~	0	10	g	a	a	c	d	a	f	g	h
h	~	~	20	~	~	~	10	0	h	a	b	c	d	e	f	g	h

11b. ábra. Mátrixmódszer első lépés

T	a	b	c	d	e	f	g	h	C	a	b	c	d	e	f	g	h
a	0	30	40	~	40	~	50	~	a	a	b	b	d	e	f	b	h
b	30	0	10	~	70	~	20	~	b	a	b	c	d	a	f	g	h
c	40	10	0	30	80	20	30	~	c	b	b	c	d	b	f	b	h
d	~	~	30	0	~	~	~	50	d	a	b	c	d	e	f	g	h
e	50	20	30	~	0	10	40	~	e	b	b	b	d	e	f	b	h
f	~	~	~	50	~	0	~	~	f	a	b	c	d	e	f	g	h
g	40	70	80	~	80	~	0	10	g	a	a	a	d	a	f	g	h
h	~	~	20	~	~	~	10	0	h	a	b	c	d	e	f	g	h

11c. ábra. Mátrixmódszer második lépés

T	a	b	c	d	e	f	g	h	C	a	b	c	d	e	f	g	h
a	0	30	40	70	40	50	50	60	a	a	b	b	b	e	e	b	b
b	30	0	10	40	70	30	20	30	b	a	b	c	c	a	c	g	g
c	40	10	0	30	80	20	30	40	c	b	b	c	d	b	f	b	b
d	70	40	30	0	110	50	60	50	d	c	c	c	d	c	c	c	h
e	50	20	30	60	0	10	40	50	e	b	b	b	b	e	f	b	b
f	120	90	80	50	160	0	110	100	f	d	d	d	d	d	f	d	d
g	40	40	30	60	80	50	0	10	g	a	h	h	h	a	h	g	h
h	50	30	20	50	90	40	10	0	h	g	c	c	c	g	c	g	h

11d. ábra. Mátrixmódszer végállapot

Mint az eddigiekből láthatjuk, az algoritmus egyszerű és *számításigénye* a pontszám harmadik hatványával jellemezhető (ld. a három ciklus). Nagy előnye, hogy a lehető *legegyszerűbb adatstruktúrákat és adatelérési formákat* (tömbök, indexezés) használja, nincs benne sem rendezés sem keresés, emiatt a gyakorlatban a nála elvben jobb algoritmusoknál is gyorsabb.

Nagy hátránya viszont, hogy a *tárigény* nagy, mert a két mátrixot tárolnunk kell az operatív tárban. Ha pl. a **T** egy elemére 4 bájtot, a **C** egy elemére 2 bájtot számítunk, akkor pl. egy 16 MB operatív tárral rendelkező PC -n kb. 1500 pontig, 32 MB esetén kb. 2000 pontig mehetünk el. Az algoritmus nagyon kihasználja, hogy a **T** és **C** ténylegesen *mátrix*

adatstruktúrát alkotnak - vagyis elemeik két index segítségével közvetlenül, egy lépésben elérhetők - így ezeket ténylegesen operatív tárbeli mátrixként kell tárolni, mert egyéb megoldásoknál (pl. lemezen tárolás, láncolt lista) az eljárás egyszerűségével együtt hatékonyságát is elveszti.

### 3.2.2. Fa-építő eljárások

#### 3.2.2.1. Alapfogalmak és alapeljárások

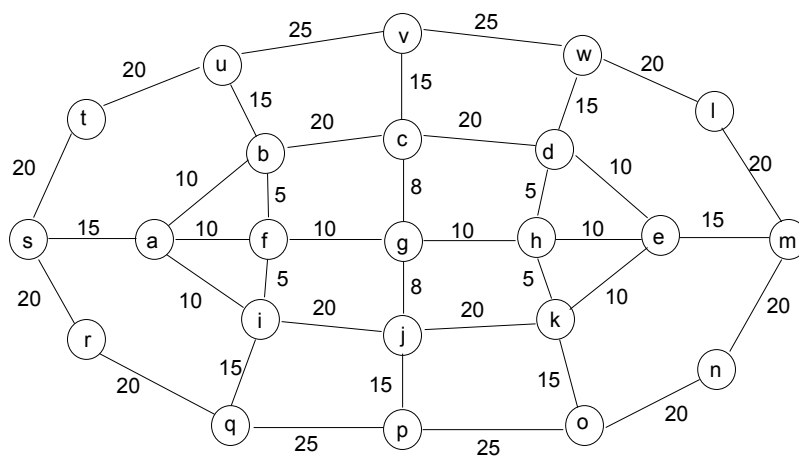
A ráterhelési feladat minimálisút-kereső feladatának tipikus, szinte kizárólagos megoldását azok az útkereső algoritmusok adják, amelyek célja és módszere a *fa-építés* (tree-building), vagyis az *egy pontból mint kezdőpontból kiinduló minimális utak fájának*, rövid elnevezéssel a *minimális fának* valamilyen konstrukciója, felépítése.

Az algoritmustípus lényeges számítástechnikai jellemzői:

- Tárigénye a szükséges minimális, két-három, egyenként pontszám méretű tömb.
- Működési módja jól illeszkedik a hálózat éltárolási adatstruktúrájához.
- Az egy fa felépítéséhez szükséges műveletek száma a pontszám négyzetével, vagy ennél alacsonyabb nagyságrenddel arányosan becsülhető.

Ezekből következően az algoritmustípus jól illeszkedik a ráterhelési feladathoz, hiszen ennél:

- Nagy, de ritka hálózatokkal dolgozunk,
- Nem minden viszonylatban, hanem csak a forrás-nyelő pontok között kell utat keresni.



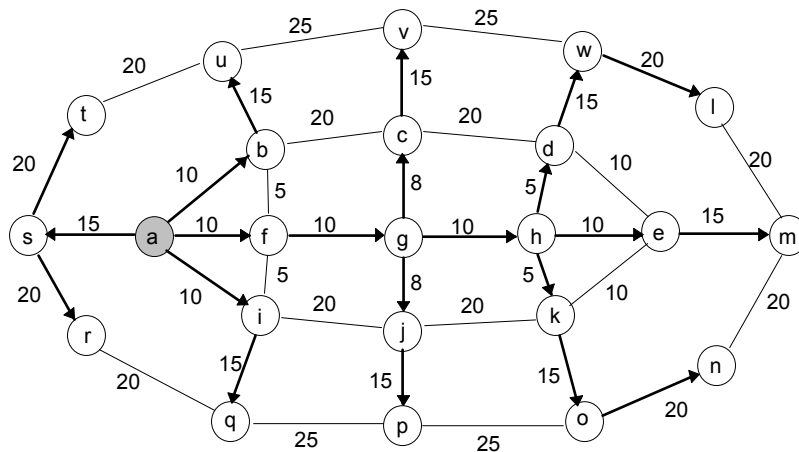
12.ábra. Fa-építő példahálózat

A fák és a faépítés szemléltetésére a 12. ábra hálózatát fogom felhasználni. Ez a hálózat csupa kétirányú, és a két irányban azonos hosszú éleket tartalmaz.

Jól ismert gráfelméleti tény (pl.[1]), hogy ha egy rögzített kezdőpontból minden más pontba kijelölünk egy (és csak egy) minimális utat, akkor egy *irányított fát* kapunk a hálózatban. Ennek gyökérpontja a kezdőpont. Ezt nevezzük a *minimális utak fájának*, rövid elnevezéssel a *minimális fának*. Mivel két pont között több azonos minimális hosszú út is lehet, a minimális fa nem feltétlenül egyértelmű, a konkrét algoritmustól függ, hogy adott esetben melyiket állítja elő.

Az irányított fa számítástechnikailag tömören tárolható és könnyen kezelhető struktúra, hiszen ha egy tömbben (a kezdőpontot kivéve) minden ponthoz megadjuk az elődjét, akkor leírtuk a fát. Ezt a tömböt a fa *címketömbjének* nevezzük. Ehhez még egy *távolságtömbben* tárolva az úthosszakat, az információ teljes.

A 13a. ábra egy **a** gyökérpontú minimális fát mutat be a hálózatban, a 13b. ábrán látható a megfelelő címke és távolságtömb, amelyeket itt is **C** és **T** jelöl. A kezdőpont címkeje csak technikai okokból önmaga.



13a. ábra. Minimális fa

	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w
C	a	a	g	h	h	a	f	g	a	g	h	w	e	o	k	j	i	s	a	s	b	c	d
T	0	10	28	35	40	10	20	30	10	28	35	70	55	70	50	43	25	35	15	35	25	43	50

13b. ábra. Címke és távolság

A faépítő algoritmusok a **T** és a **C** egy kezdőállapotából kiindulva, a tömböket - tehát a fát - lépésenként építve és korrigálva, több lépés megtétele után jutnak el a végeredményhez. Az eljárások leírásához, a **T** és a **C** mellett vezessünk be még néhány jelölést, amelyek a faépítés folyamatában értelmeződnek és használtak:

- **A P** a hálózati pontok halmaza, a további halmazok ennek részhalmazai.
- Az **A** az aktív pontok halmaza, elemei már rendelkeznek távolsággal és címkével de ezek még változhatnak.
- **A K** a kész pontok halmaza, elemei már végleges (nem rövidíthető) távolsággal és emiatt végleges címkével is rendelkeznek.
- Az **a** pont a felépítendő fa gyökérpontja vagyis az utak kezdőpontja, egyben az építési eljárás kiindulópontja.

A faépítés elvi alapeljárása az alábbi:

- **Kezdőállapot**  
Rendeljük a kezdőponthoz a 0, a többihez a végtelen távolságértéket, az **A** tartalmazza csak a kezdőpontot, tehát  $C[a] = a, T[a] = 0, A = \{a\}$ . Ez tehát az a fa, amely egy pontból, a gyökérpontból áll.

- *Javító lépések*
  - a) Valamilyen (előre rögzített eljárás, választási stratégia) szerint válasszuk ki az  $\mathbf{A}$  egy elemét, jelölje ezt  $\mathbf{x}$ , ezt töröljük az  $\mathbf{A}$ -ból,  $\mathbf{A}=\mathbf{A}-\{\mathbf{x}\}$ .
  - b) Az  $\mathbf{x}$  -ből *kiinduló minden élre*, tehát minden  $(\mathbf{x}, \mathbf{y})$  élre végrehajtandó: Megvizsgáljuk, hogy  $\mathbf{y}$  útja  $\mathbf{x}$ -en keresztül rövidíthető-e. Ha igen, a pont címkéjét  $\mathbf{x}$  -re, távolságát a rövidebbre állítjuk, és hozzávesszük az  $\mathbf{A}$  halmazhoz. Tehát:
    - b1) Ha  $\mathbf{y}$  még nem volt a fában, akkor  $\mathbf{x}$  előddel bekerül.
    - b2) Ha  $\mathbf{y}$  már bent volt a fában, akkor új elődje az  $\mathbf{x}$  lesz.

Képletben: ha  $\mathbf{T}[\mathbf{y}] > \mathbf{T}[\mathbf{x}] + \mathbf{h}(\mathbf{x}, \mathbf{y})$   
akkor  $\mathbf{T}[\mathbf{y}] = \mathbf{T}[\mathbf{x}] + \mathbf{h}(\mathbf{x}, \mathbf{y})$ ,  $\mathbf{C}[\mathbf{y}] = \mathbf{x}$ ,  $\mathbf{A} = \mathbf{A} + \{\mathbf{y}\}$ .

  - c) Ha van még aktív elem, vagyis ha az  $\mathbf{A}$  nem üres, akkor folytatjuk az a) ponttól.
- *Végállapot*  
Ha már nincs aktív pont, akkor az eljárás véget ért, a minimális fa készen van, a  $\mathbf{C}$  és  $\mathbf{T}$  meghatározzák. Az egyes végpontokhoz tartozó utak a  $\mathbf{C}$  alapján (a végpontból visszafelé haladva) egyszerűen összerakhatók. Ha valamely  $\mathbf{z}$  ponthoz a  $\mathbf{C}[\mathbf{z}]$  nem definiált, vagyis a  $\mathbf{T}[\mathbf{z}]$  végtelen maradt, akkor az  $\mathbf{x}$ - $\mathbf{z}$  viszonylatban nincs út.

Egy egyszerű, szemléletes, matematikailag könnyen kezelhető választási stratégia, amely az alkalmazásokban is hatékonyan bizonyult, és a faépítő algoritmusok egy nagy családját definiálja:

*Válasszunk ki egy minimális távolságú aktív pontot.*

Az ilyen algoritmusokat a szakirodalom a címkekezelés módjáról *címke-beállító* (label setting) illetve az első publikálóról [20] *Dijkstra*-féle eljárásoknak nevezi.

A minimum-elv bevezetésével az alapalgoritmus módosítása, vagyis a *Dijkstra*-féle faépítő elvi alapeljárás a következőképpen fogalmazható meg:

- *Kezdőállapot*  
Rendeljük a kezdőponthoz a 0, a többihez a végtelen távolságértéket, a  $\mathbf{K}$  legyen üres, az  $\mathbf{A}$  tartalmazza csak a kezdőpontot, tehát  $\mathbf{C}[\mathbf{a}] = \mathbf{a}$ ,  $\mathbf{T}[\mathbf{a}] = 0$ ,  $\mathbf{K} = \{\}$ ,  $\mathbf{A} = \{\mathbf{a}\}$ . Ez tehát az a fa, amely egy pontból, a gyökérpontból áll.
- *Javító lépések*
  - a) Válasszuk ki az  $\mathbf{A}$  egy minimális távolságú elemét, jelölje ezt  $\mathbf{x}$ , ezt töröljük az  $\mathbf{A}$ -ból, és vegyük hozzá a  $\mathbf{K}$ -hoz. Képletben:  $\mathbf{T}[\mathbf{x}] \leq \mathbf{T}[\mathbf{u}]$  minden  $\mathbf{A}$ -beli  $\mathbf{u}$  elemre,  $\mathbf{A} = \mathbf{A} - \{\mathbf{x}\}$ ,  $\mathbf{K} = \mathbf{K} + \{\mathbf{x}\}$ .
  - b) Nem változik, azonos az általános eljárással.
  - c) Nem változik, azonos az általános eljárással.
- *Végállapot*  
Nem változik, azonos az általános eljárással.

Az algoritmus első néhány lépését a példahálózaton a 14. ábrák mutatják be. A konkrét példában az  $\mathbf{A}$  aktivitáshalmaz elemeit egy - a távolság szerint növekvően rendezett - lineáris adatszerkezetben (tömb vagy lista) tartjuk, az új aktív pontot eszerint soroljuk be, ennek következtében a  $\mathbf{K}$  kész halmazba mindig az első aktív pont kerül át.

Megjegyzendő, hogy a  $\mathbf{K}$  halmaz csak a leírást és megértést (valamint a helyességbizonyítást) könnyíti, a számítástechnikai realizációkból el is hagyható.



- Az algoritmus minden javító lépésében egy pont bekerül a  $\mathbf{K}$  halmazba, tehát pontosan  $\mathbf{N}$  lépés szükséges.
- Minden javító lépésben lezajlik egy, maximum  $\mathbf{N}-1$  elem közti minimumkeresés ( $\mathbf{A}$  halmaz) és maximum  $\mathbf{N}-1$  élre ( egy pontból kiinduló élek) vonatkozó rövidítési vizsgálat.

A példa (14. ábra) alapján is könnyen elképzelhető, hogy a *közlekedési hálózatoknál* a fenti, az általános esetre vonatkozó becslésnél jobb a helyzet, hiszen amellett, hogy az  $\mathbf{N}$  lépés természetesen szükséges, ezen belül:

- Az  $\mathbf{A}$  halmaz átlagos elemszáma viszonylag kicsi, nagy hálózatoknál az  $\mathbf{N}$  -értéknek csak néhány százaléka.
- Egy pontból az  $\mathbf{N}-1$  értéknél lényegesen kisebb számú él indul ki átlagosan.

Könnyen belátható az is, hogy bármilyen faépítő algoritmust is konstruálunk, a hálózat *minden élet legalább egyszer* meg kell vizsgálni, és ezt a Dijkstra algoritmus élenként *pontosan egyszer* teszi meg. Következésképpen, az ebből az alapalgoritmusból származtatott, ennek alapelveit (egyszeri élvizsgálat, *kész* halmaz, *aktív* halmaz, *minimumkiválasztás*) követő algoritmusok hatékonysága az alábbi tényezőkön múlik:

- *Az aktivitás halmaz elemszáma.*
- *Az aktivitás halmaz tárolási, kezelési módja, az e célra választott adatstruktúra.*

Az aktivitáshalmaz számítástechnikai reprezentációi az alábbi három típusba oszthatók:

- $\alpha$  : *rendezetlen* (halmaz, rendezetlen tömb, lista)
- $\beta$  : *lineárisan rendezett* (tömb, lista)
- $\gamma$  : *többszintű ill. részben rendezett* (összetett listák, fák, kupacok stb.)

A rendezettség természetesen mindig a *pont távolsága szerinti növekvő* (nemcsökkenő) rendezettséget jelent.

A vonatkozó szakirodalmat áttekintve látható, hogy a hatékonyság javítására irányuló erőfeszítések elsősorban az aktivitás halmaz tárolási módjára irányultak, konstruálva olyan - a fenti tipizálás szerint általában a  $\gamma$  csoportba tartozó - adatszerkezeteket, amelyekkel a lépésszám csökkenthető. Néhány jellegzetes ilyen publikációt láthatunk a [41], [18], [16] hivatkozásokban. Ezekkel a módszerekkel a hatékonyság fent említett négyzetes becslése lényegesen javítható ( $N^2$  helyett  $N * \log N$  ). Az alábbiakban a javítást egy más módon keressük, az aktivitás halmaz méretének csökkentésére alapozva.

Az algoritmusnak az aktivitás halmazt kezelő javító lépéseit elemezve látható:

- a) lépés: a *minimumkiválasztás* az  $\alpha$  típusban egy tényleges, az elemszámmal egyenesen arányos lépésszámú minimumkeresést jelent, a másik két típusban a minimum közvetlenül adódik (első elem, csúc elem). A *törlés* az aktivitáshalmazból az első két típusban nem műveletigényes, a  $\gamma$  típusban viszont a legtöbb esetben a struktúra törlés utáni helyreállítása legalább ugyanolyan műveletigényes feladat mint a besorolás.
- b) lépés b1) eset: ez az  $\alpha$  típusban nem műveletigényes, a másik kettőben viszont egy új elem *besorolását*, vagyis strukturábani helyének megkeresését és beillesztését jelenti, ami az elemszám valamilyen (a  $\beta$  típusban lineáris, a  $\gamma$  -ban általában logaritmikus) függvénye.

- b) lépés b2) eset: a műveletigény azonos jellegű mint a b1) esetben, azzal az eltéréssel, hogy a  $\beta$  és  $\gamma$  típusban átsorolásról, vagyis egy már a struktúrában lévő pont új helyének megkereséséről és áthelyezéséről van szó.

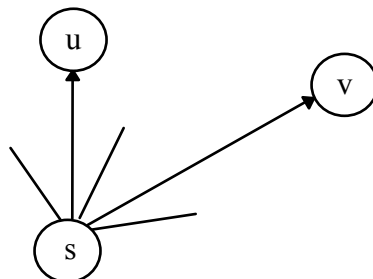
*Az elemzésből levonható az a következtetés, hogy az aktív elemek számának csökkentése minden típusban javulást eredményez az algoritmus hatékonyságában. A cél egy ilyen módszer kifejlesztése [49].*

### 3.2.2.2. Egy új eljárás

A javítás lehetőségére az utalt, hogy konkrét példákat vizsgálva úgy tűnt, hogy a közlekedési hálózatoknál az *aktivitás halmaz* túl bő, sok olyan pont van, amely a halmazba való bekerüléséhez képest csak *több lépésnyi várakozás* után kerül a minimumpozícióba.

*Célszerű lenne tehát egy-egy lépésben kevesebb új pontot bevonni, elsősorban olyanokat, amelyek nagyobb eséllyel juthatnak a minimumpozícióba, de természetesen gondoskodva arról, hogy mindig bent legyen az elméletileg következő minimumpont.*

Az eljárás alapötletét az a szemléletesen világos állítás adja, hogy *az egy pontból kiinduló élek csak hosszuk szerint növekvő (nemcsökkenő) sorrendben kerülhetnek be a minimális fába*, az ellenkező eset nem lehetséges, így a vizsgálatnak is célszerű ezt a sorrendet követnie. A 15. ábra példáján: ha  $h(s, u) < h(s, v)$ , akkor az  $(s, v)$  él nem előzheti meg az  $(s, u)$  élt a bekerülésnél.



15. ábra. Bekerülési sorrend

Erre egy kedvező lehetőséget ad, ha a hálózat éleit, pontosabban pontonként a kiinduló éleket, növekvő (nemcsökkenő) hossz szerint átrendezzük. Ez az *előrendezés a közlekedési hálózatok ráterhelési feladatánál* nem okoz lényeges többlet-műveletigényt (vagyis a hatékonyság csökkenését), hiszen:

- az egy pontból kimutató élek száma konstans módon korlátozott, valamint
- egy előrendezés elég az összes minimális fához,

ennek következtében az előrendezés műveletigényének az egy fára jutó hányada, a fa előállításához szükséges műveletek számához képest elhanyagolhatóan kicsi.

A *rendezettség* nyújt lehetőséget arra, hogy egy pont *kész* állapotba kerülésekor kevesebb új pont lépjen be az *aktivitás halmazba*, de természetesen gondoskodnunk kell arról, hogy sohasem hiányozhasson a következő minimum pontja.

Az egyértelműség kedvéért úgy rendezünk, hogy ha két él azonos hosszú, akkor az kerüljön előre, amelyiknek kisebb a sorszámja. Az egy pontból kimutató élek sorrendjének

nyilvántartására és kezelésére vezessünk be egy  $\mathbf{M}$  mutatótömböt. Az algoritmus végrehajtása folyamán az  $\mathbf{M}[\mathbf{i}]$  jelenti az  $\mathbf{i}$  pontból kimutató, soron következő, még nem vizsgált él sorszámát.

- *Kezdőállapot*  
Rendeljük a kezdőponthoz a 0, a többihez a végtelen távolságértéket, a  $\mathbf{K}$  legyen üres, az  $\mathbf{A}$  tartalmazza csak a kezdőpontot, tehát  $\mathbf{C}[\mathbf{a}] = \mathbf{a}, \mathbf{T}[\mathbf{a}] = 0, \mathbf{K} = \{\}, \mathbf{A} = \{\mathbf{a}\}$ . Minden  $\mathbf{i}$ -pontra, amelyből indul ki él, legyen az  $\mathbf{M}[\mathbf{i}] = 1$ .
- *Javító lépések*
  - a) Válasszuk ki az  $\mathbf{A}$  egy minimális távolságú elemét, jelölje ezt  $\mathbf{i}$ , ezt töröljük az  $\mathbf{A}$ -ból, és vegyük hozzá a  $\mathbf{K}$ -hoz. Jelölje az  $\mathbf{i}$  címkéjét  $\mathbf{e}$ . Képletben:  $\mathbf{T}[\mathbf{i}] \leq \mathbf{T}[\mathbf{u}]$  minden  $\mathbf{A}$ -beli  $\mathbf{u}$  elemre,  $\mathbf{A} = \mathbf{A} - \{\mathbf{i}\}, \mathbf{K} = \mathbf{K} + \{\mathbf{i}\}, \mathbf{e} = \mathbf{C}[\mathbf{i}]$ .
  - b) Mind az  $\mathbf{e}$ , mind az  $\mathbf{i}$  pontra hajtsuk végre az alábbi (b1-b4) belső eljárást:
    - b1) Jelölje a pontot  $\mathbf{x}$ .
    - b2) Az  $\mathbf{M}[\mathbf{x}]$  által mutatott éllel indulva, a  $\mathbf{x}$ -ből kimutató éleket egyenként sorra véve (és természetesen a mutatót léptetve) eljutunk az első olyan  $(\mathbf{x}, \mathbf{y})$  élhez, amelyre a  $\mathbf{T}[\mathbf{x}] + \mathbf{h}(\mathbf{x}, \mathbf{y}) < \mathbf{T}[\mathbf{y}]$  rövidítési lehetőség fennáll, vagy már nincs több, még nem vizsgált  $\mathbf{x}$  kezdőpontú él. Az utóbbi esetben a belső eljárásnak vége, az előbbi esetben folytatjuk a b3) ponton.
    - b3) Ha az  $\mathbf{y}$  még nem aktív, hozzá vesszük az  $\mathbf{A}$ -hoz, címkéjét, távolságát módosítjuk:  $\mathbf{C}[\mathbf{y}] = \mathbf{x}, \mathbf{T}[\mathbf{y}] = \mathbf{T}[\mathbf{x}] + \mathbf{h}(\mathbf{x}, \mathbf{y})$ , az  $\mathbf{x}$  mutatóját léptetjük:  $\mathbf{M}[\mathbf{x}] = \mathbf{M}[\mathbf{x}] + 1$ , és a belső eljárásnak ezen a ponton is vége van.  
Ha az  $\mathbf{y}$  már aktív volt, folytatjuk a b4) ponton.
    - b4) Mivel az  $\mathbf{y}$  aktív, van címkéje, jelölje ezt  $\mathbf{z}$ , tehát  $\mathbf{z} = \mathbf{C}[\mathbf{y}]$ . Mint az előző pontban:  $\mathbf{C}[\mathbf{y}] = \mathbf{x}, \mathbf{T}[\mathbf{y}] = \mathbf{T}[\mathbf{x}] + \mathbf{h}(\mathbf{x}, \mathbf{y}), \mathbf{M}[\mathbf{x}] = \mathbf{M}[\mathbf{x}] + 1$ . A  $\mathbf{z}$  ponttal ismételjük meg a vizsgálatot, tehát legyen  $\mathbf{x} = \mathbf{z}$  és menjünk vissza a b2) pontra.
  - c) Ha van még aktív elem, vagyis ha az  $\mathbf{A}$  nem üres, akkor folytatjuk a b) ponttól.
- *Végállapot*  
Nem változik, azonos az általános eljárással.

Az algoritmus első néhány lépését a példahálózaton a 16. ábrák mutatják be. Az aktivitáshalmaz tárolási és kezelési módja megegyezik a 14. ábrákon mutatott alapeljárásával, így a két eljárás a példán is könnyen összehasonlítható.

Itt nem részletezendően, az [52] és [56] cikkekben egzakt módon bebizonyítom, hogy az új eljárás matematikailag *helyes*, tehát minimális fát állít elő, valamint az is, hogy a két algoritmus minden lépésében a módosított algoritmus *aktivitás halmaza része az eredeti aktivitás halmazának*, vagyis az aktivitás halmaz elemszáma lépésenként is alatta marad az eredetiének.

Az, hogy az *aktivitás halmaz* halmaz mérete, és ezzel a *kiválasztási-besorolási-átsorolási* lépések száma ténylegesen mennyivel csökken, *függ* egyrészt a konkrét *hálózattól*, másrészt a halmaz *tárolási-kezelési* módjától. A függelékben néhány konkrét hálózattal és algoritmussal demonstrálom azt, hogy a *javítás tényleges*, a *hatékonyság* a módszer alkalmazásával *nő* (függelék 2.1 pont).



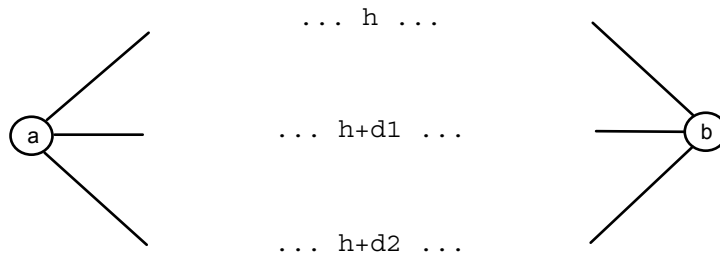


### 3.3. További utak

#### 3.3.1. A feladat megfogalmazása, alapfogalmak

A forgalomráterhelési feladat megfogalmazásánál mondtuk ki, mint az egyensúlyi modellre vonatkozó egyik fontos alapfeltevést: *Az utazó, hogy célját elérje, a lehetséges útvonalak közül a minimális költségű útvonalat választja.*

Azonban már a fentebbi - minimális utak meghatározására vonatkozó - fejtegetéseinkből is következik, hogy ez a megfogalmazás így önmagában nem feltétlenül egyértelmű, pontosításra szorul. Mint láttuk, nem zárhatók ki az *azonos minimális költségű* utak és könnyen elképzelhető, hogy valamely konkrét közlekedési hálózat valamely viszonylatában több olyan útvonal van, amelyek a hálózati modellben ugyan eltérő költségűek, de ez az eltérés a *valós hálózaton*, a valós utazó által *nem érzékelhető*, lényegtelen. Ha ezt a tényt nem vesszük figyelembe, hibát követhetünk el modellünk alkalmazásánál.



17. ábra. Közel azonos költségű utak

Megjegyzem, hogy dolgozatomban kiemelten tárgyalt modellünk a *lépcsős ráterhelés* ebből a szempontból szerencsésnek mondható, hiszen - ellentétben más modellekkel - maga az eljárás kiküszöböli, kijavítja az ilyen típusú hibákat. Az eljárás lényegi tulajdonsága, hogy az egyes lépcsőkben végrehajtott ráterhelésekkel a terhelt hálózati elemek ellenállása nő, a minimális utak szükségszerűen az előző lépcső minimális utjaitól a *legkevesbé eltérő költségű* (eddig még egyáltalán nem, vagy kevésbé terhelt) *útvonalakra helyeződnek át*. Ha a lépcsőzésünk nem túl nagyléptékű, nem maradnak üresen a minimálishoz költségben közel álló útvonalak. A 17. ábra példáján: ha az **a** - **b** viszonylatban a  $h$  hosszú út a minimális, a  $d_1$  és  $d_2$  viszonylag kicsi nemnegatív számok, és legalább három lépcsőben terhelünk rá, akkor mind a három út kap terhelést.

Mindazonáltal a *második, harmadik ...* (általánosan szólva) *k-adik* minimális utak megkeresésének algoritmusai csatlakoznak témánkhoz, mint az útvonalválasztás modellezésének további lehetséges eszközei. Ha a lépcsős ráterhelésben alkalmazzuk őket - a lépcsők számának csökkentése illetve a forgalom egy lépcsőn belüli szélesebb terítése céljából - akkor természetesen a modellhez definiálnunk kell egy *megosztó függvényt*, amely megmondja azt, hogy egy adott viszonylatban előállítható  $h_1 \leq h_2 \leq \dots \leq h_k$  hosszú *első, második ... k-adik* utak között a viszonylatban fellépő  $f$  forgalmat milyen  $f_1 \geq f_2 \geq \dots \geq f_k$  részekre osszuk szét, természetesen az  $f = f_1 + f_2 + \dots + f_k$  feltétel fennállása mellett.

Megosztó függvényként a szakirodalomban már klasszikusnak számító [11] általánosan javasolt alak:

$$f_r = f * \frac{(1 / h_r)^\alpha}{\sum_{j=1}^k (1 / h_j)^\alpha} \quad 1 \leq r \leq k$$

ahol  $k$  a figyelembe vett utak száma.

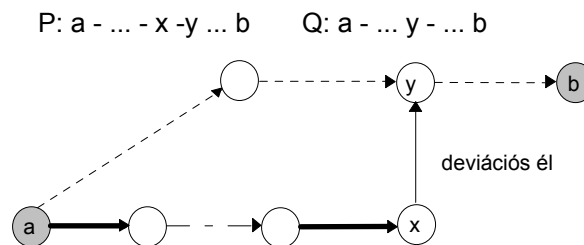
A képlet azt a tapasztalati tényt fejezi ki, hogy a forgalomból való részesedés a hossz növekedésével exponenciálisan csökken. A változás mértékét az  $\alpha$  kitevő adja meg, amely az útvonalválasztó algoritmusnak (a tervező által megadott) paramétere, és értékére általában a  $2 \leq \alpha \leq 8$  reláció áll fenn [11].

A  $k$ -adik minimális utak kutatása az alapproblémaéval (minimális utak) együtt indult [25], [14]. Viszonylag egyszerű és gyors algoritmusok készíthetők az ún *deviációs* elvre alapozva (kimondva és bizonyítva [25] -ben):

Egy  $a - b$  viszonylatra vonatkoztatva, egy  $Q$  út egy deviációján értünk egy olyan  $P$  utat, amely kezdetben a minimális fában halad, ezután egy éllel - ezt *deviációs élnek* nevezzük - becsatlakozik a  $Q$  útba (18. ábra). (A pontosság kedvéért : a deviációs él nem lehet a  $Q$  része.)

Legyen a  $P$  egy út az  $a - b$  viszonylatban:

- Ha a  $P$  nem minimális, akkor van olyan  $Q$  út a viszonylatban, amelynek a  $P$  egy deviációja és a  $Q$  hossza kisebb mint a  $P$  hossza.
- Ha a  $P$  minimális, akkor vagy az egyetlen minimális út a viszonylatban, vagy egy másik minimális út egy deviációja.

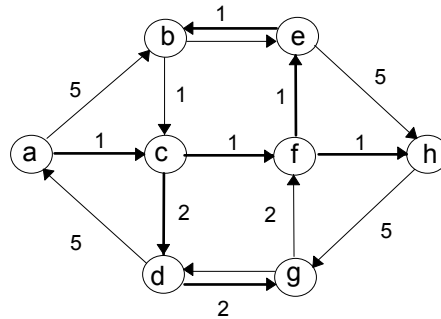


18. ábra. Deviáció

A deviációk képzésével könnyen generálhatjuk a *második, harmadik ... k - adik* utakat (számítástechnikai szempontból persze nem egyszerű probléma ezek regisztrálása, tárolása), de könnyen észrevehető a témakör alapvető problémája, hogy a *nem minimális utak nem feltétlenül körmentesek* (ld. 19.ábra, a vastagított élek az  $a$  gyökérpontú minimális fát mutatják, az  $a - h$  viszonylat második legrövidebb útja - amely az  $a - c - f - e - b - c - f - h$  útvonal - körös). Viszont az alkalmazások többségében, így a ráterhelésnél is igaz az, hogy a körös utak nem értelmezhetőek a modell szempontjából.

A körmentesség biztosítása, illetve a körös utak kiszűrése *minőségileg* új követelmény, amely nagyon nagy mértékben megnöveli a számításgigényt. Több és többfajta algoritmus, illetve számítástechnikai implementáció ismert [75], [74], [68], [28], de ezek közös jellemzője (mint

ahogy ez az áttekintő jellegű [63] cikkben is megállapított), hogy *legalább egy nagyságrenddel* több számítás szükséges, mint a minimális utaknál. Pontosabban szólva, míg a minimális utaknál az egy viszonylatra jutó számításigény nagyságrendben  $\log N$  és  $N$  között van, ugyanez az érték itt  $N^2$  és  $N^3$  közé esik.



19. ábra. Körös második út

Összevetve ezt a fentebbiekkel, az a következtetés vonható le, hogy a lépcsős ráterhelésnél legfeljebb csak a  $k$  kisebb értékeire javasolt a  $k$ -adik utak keresése. Az alábbiakban egy - korábban publikált [42], [43] algoritmusból továbbfejlesztett - a deviációs elven alapuló saját algoritmust adok a  $k=2$  esetre.

### 3.3.2. Egy második utas eljárás

Az algoritmus (a fa építéshez hasonlóan) az *egy pontból* mint kezdőpontból *kiinduló második legrövidebb utakat* határozza meg. Felhasználja a kezdőponthoz mint gyökérponthoz tartozó minimális fát. Az algoritmus leírásánál *viszonylaton* mindig egy olyan viszonylatot értünk, amelynek kezdőpontja a gyökérpont.

A deviációs elvből következik [25], hogy ha rögzítünk egy minimális fát, és ha valamely viszonylatnak van második minimális útja (nem csak egyetlen út vezet a kezdő és végpont között), akkor ez *pontosan egy* olyan élt tartalmaz, amely nincs benne a minimális fában. Az algoritmus ezt az élt keresi meg, egy iterációs eljárással.

Az algoritmus leírásához vezessük be a következő fogalmakat:

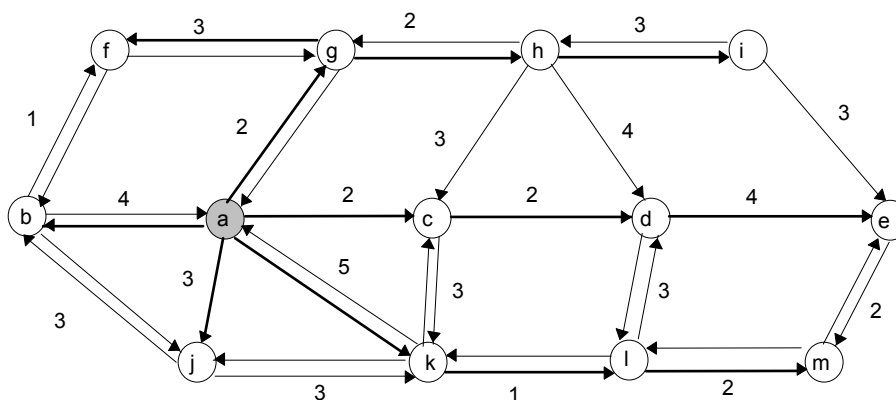
- *Kerülő* útnak nevezünk egy olyan utat, amely *pontosan egy* olyan élt tartalmaz, amely nincs benne a minimális fában. Ezt az élt, az útra vonatkozóan *kerülő élnek* nevezzük.
- *Végponti kerülő út* az olyan kerülő út, amelynek *utolsó* éle a kerülő él.
- *Minimális végponti kerülő út* a minimális hosszú végponti kerülő út.

Az algoritmus - a minimális fa meghatározásához hasonlóan - egy címkézési technikával dolgozik, ehhez vezessük be a **C2** és **T2** *másodlagos* címke és távolságtömböt. Az egységes jelölés kedvéért az eredeti - a minimális fához tartozó - ilyen tömböket itt jelölje **C1** és **T1**. A rögzített kezdőpontot jelölje **a**.

- *Kezdőállapot*  
A minimális fa, tehát a **C1** és **T1** adott. A **C2** és **T2** - t illetően rendeljük a kezdőponthoz a 0, a többihez a végtelen távolságértéket. Ezután az **a** pontot kivéve, minden **x** ponthoz mint végponthoz határozzuk meg a minimális végponti kerülő utat. Ennek hosszát vegye fel a **T2 [x]**, a **C2 [x]** értéke pedig legyen ezen út kerülő élének kezdőpontja.

- *Javító lépések (iteráció)*
  - a) Az **a** -tól különböző minden **y** pontra megvizsgáljuk, hogy **y** kerülő útja az őt a minimális fában megelőző  $\mathbf{x}=\mathbf{C1}[\mathbf{y}]$  ponton keresztül rövidíthető-e. Ha igen, a pont másodlagos címkéjét **x** -re, távolságát a rövidebbre állítjuk. Képletben: ha  $\mathbf{T2}[\mathbf{y}] > \mathbf{T2}[\mathbf{x}] + \mathbf{h}(\mathbf{x}, \mathbf{y})$  akkor  $\mathbf{T2}[\mathbf{y}] = \mathbf{T2}[\mathbf{x}] + \mathbf{h}(\mathbf{x}, \mathbf{y})$ ,  $\mathbf{C2}[\mathbf{y}] = \mathbf{x}$ .
  - b) Ha volt olyan pont, amelynél történt változás, akkor ismételjük az a) részlépést, ha ilyen nincs, készen vagyunk.
- *Végállapot*  
 A **C2** és **T2**, a **C1** segítségével meghatározzák a második utakat. Legyen a **z** egy végpont. A  $\mathbf{T2}[\mathbf{z}]$  adja a **z** második útjának hosszát. Maga az út (a végpontból visszafelé haladva) úgy rakható össze, hogy a  $\mathbf{C2}[\mathbf{z}]$  -vel indulva addig haladunk a **C2** szerint (a minimális fában visszafelé), amíg el nem érjük a kerülő élt (az első él, amely nincs a minimális fában). A kerülő él kezdőpontjával kezdve átváltunk a **C1** címketömbre, és ebben haladunk a kezdőpontig. Ha valamely **z** ponthoz a  $\mathbf{C2}[\mathbf{z}]$  nem definiált, vagyis a  $\mathbf{T2}[\mathbf{z}]$  végtelen maradt, akkor az **a-z** viszonylatban nincs második út.

Az algoritmus működését egy példahálózaton a 20. ábrák mutatják be. A 20a. ábrán vastagítva jelöltem a minimális fába eső éleket.



20a. ábra. Második utak

	a	b	c	d	e	f	g	h	i	j	k	l	m
<b>C1</b>	a	a	a	c	d	g	a	g	h	a	a	k	l
<b>T1</b>	0	4	2	4	8	5	2	4	7	3	5	6	8

0	a	b	c	d	e	f	g	h	i	j	k	l	m
<b>C2</b>	a	f	h	h	i	b	h	i		b	c	d	e
<b>T2</b>	0	6	7	8	10	5	6	10	~	7	5	7	10

1	a	b	c	d	e	f	g	h	i	j	k	l	m
<b>C2</b>	a	f	h	h	i	b	h	g	h	b	c	k	l
<b>T2</b>	0	6	7	8	10	5	6	8	11	7	5	6	8

2	a	b	c	d	e	f	g	h	i	j	k	l	m
<b>C2</b>	a	f	h	h	i	b	h	g	h	b	c	k	l
<b>T2</b>	0	6	7	8	10	5	6	8	11	7	5	6	8

a	c	k	l	m			8
a	g	h	i	e			10
a	g	h	g	h	i		11

20b. ábra. Második utak

A példánál 1 iterációs lépés elég a második utak beállításához. A 20b. ábra utolsó táblázata néhány ilyen utat tartalmaz. Ebből is látható, hogy ugyanúgy mint más, az egyszerű deviációképzésen alapuló módszerek, ez sem garantálja a körmentességet.

Ha egy viszonylatban *körös* utat kaptunk, valamilyen egyszerű javítást alkalmazunk. Vehetjük a második út *közelítéseként* a *körmentes kerülő* utak közül a minimális hosszút, vagy magát a *minimális utat* is. Előrebocsátom, hogy erre a korrekcióra valós hálózatoknál csak jelentéktelen számban van szükség.

Eltekintve most a körmentesség problémájától, vizsgáljuk meg az eljárás *hatékonyságát*. Látható, hogy *tárigénye* csak egy újabb címke és távolságtömb, tehát a minimálisan szükséges tárral dolgozik.

A számításigényt vizsgálva adódik, hogy a minimális végponti kerülő utak beállításához - adott minimális fa esetén - elegendő élenként egy vizsgálat (az él lehet-e kerülő él, és a végpontjára vonatkozóan rövidít-e). Egy iterációs lépésben minden pontot egyszer kell megvizsgálni. Eredményként megkapjuk az összes, a kezdőpontból induló második utat. Következésképpen általános hálózatnál az egy kezdőpontra jutó számításigényt nagyságrendileg az  $N^2 + r * N$  képlet jellemzi, ahol  $r$  az iterációk száma. Ez egy útra vetítve  $N + r$  nagyságrendű. Következésképpen közlekedési hálózatnál, ahol az élek száma kiindulópontonként erősen korlátozott, az egy viszonylatra jutó számításigényt alapvetően az iterációk száma határozza meg. Az iterációk számának elvi felső korlátja a pontok száma.

Célszerű tehát megvizsgálni, hogy valós közlekedési hálózatoknál hogyan alakul az *iterációk száma* és a körös utak aránya. Erre vonatkozóan több hálózaton végeztem kísérleteket, számszerű eredményeket a függelék 2.2. pontban adok. Ezek szerint:

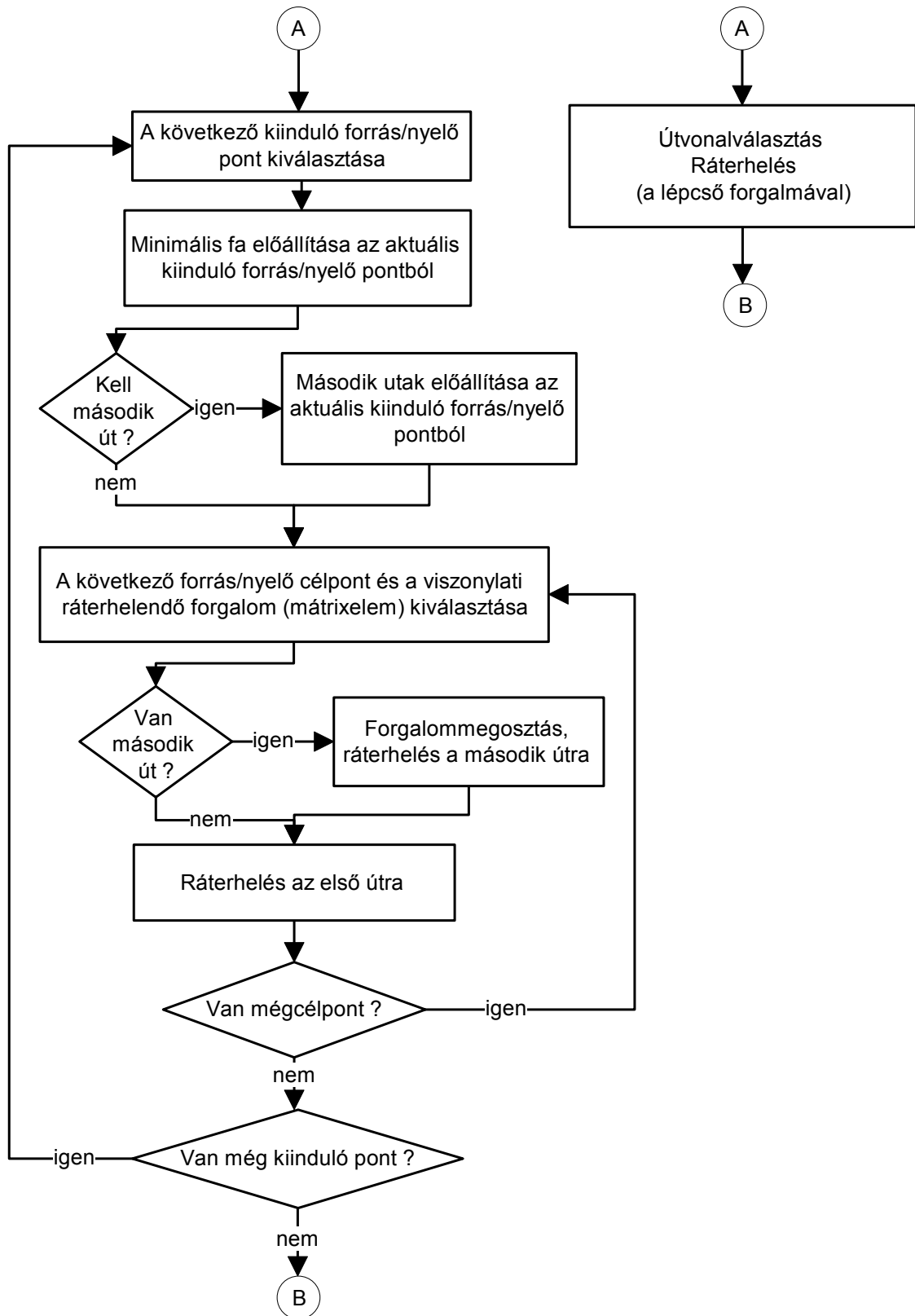
- *A valós közlekedési hálózatoknál a körös második út nem gyakorlati eset, az ilyen kivételek feldolgozásának számításigénye a teljes számításigényhez képest elhanyagolható.*
- *Az iterációk száma az elvi maximumhoz (a pontok számához) képest nagyon kicsi, a hálózat méretével csak nagyon lassan, a logaritmikus növekedéshez hasonlóan nő.*

Összevetve ezt a korábban a minimális utakra vonatkozó becslésekkel és a gyakorlati alkalmazásoknál megfigyelt időeredményekkel, az a következtetés vonható le, hogy a fenti második utas eljárás alkalmazásának időigénye *nagyságrendben nem tér el* a minimális utakétól, tehát két úttal számolni a ráterhelésben ezzel a módszerrel csak kétszer annyira időbe kerül, mint egy úttal.

### 3.4. Beillesztés a ráterhelési eljárásba

Az általános lépcsős ráterhelési eljárás *útvonalválasztás-ráterhelés* blokkját a jelen fejezet ismeretanyagával bővítve a 21. ábra szerint részletezem ki.

Külön is megjegyzem, hogy az útvonalak - akár mint címke ill. távolságtömbök, akár mint pont vagy élsorozatok - csak arra a célra jönnek létre az eljárásban, hogy a forgalmat ráterheljük, vagyis az élek megfelelő jellemzőjét a megfelelő mennyiséggel növeljük. Tehát az utak nem tárolódnak. Tárolásuk nagy tárcapacitást kötné le, és a lépcsős ráterhelésnél felesleges is lenne, hiszen a következő lépcsőben, más ellenállásértékek mellett, az utak is áthelyeződnek.



21. ábra. Útvonalválasztás a ráterhelésben

## 4. Speciális modellek

### 4.1. Bevezetés

A 2. fejezetben írtuk le a forgalomráterhelési feladat általam definiált és alkalmazott hálózati *alapmodelljét*. Ez tartalmazza mindazokat a hálózati elemeket, amelyek - elméleti megfontolások és gyakorlati tapasztalatok alapján - minden alkalmazáshoz szükségesek és az általában elvárt eredményeket szolgáltatják.

Jelen fejezetben a modell bizonyos *specialitásaival* foglalkozom. Az alkalmazások gyakran megfogalmazznak olyan elvárásokat, támasztanak olyan igényeket a ráterhelési eljárással szemben, amelyek az általános modellen belül nem teljesíthetők, a modell bővítését igénylik. Ilyenek lehetnek például a tipikusan városi közlekedési hálózatok sajátos jellegzetességeiből adódó igények.

A jelen időszakban (a kilencvenes években) kiemelt feladatcsoport a magyarországi *autópályahálózat* bővítésének tervezése. Ebből adódóan a hálózati modellekben is meg kell jeleníteni az *autópályát* mint speciális kezelést igénylő *részhálózatot*, amely az általános esethez képest több és többféle jellemzővel is bír, de a bonyolultabb modell több és többféle eredményt is ad [37]. Ilyen jellemző például az autópályadíj és nagyon hasznos eredmény a meglévő vagy tervezett autópályák *díjérzékenységi* vizsgálata vagyis a várható forgalmi terhelés, és ezzel a *díjbevétele* előrebecslése a díjszintek függvényében.

A következőkben speciális eseteket tárgyalok, megadva az általam kidolgozott és javasolt *megoldásokat* az alábbi problémákra:

- A forgalom *kitiltása* egyes hálózatrészekről.
- Egyes hálózatrészekre vett *relatív forgalmak* előállítás.
- Az *autópályadíjak* figyelembevétele, többfajta útdíjképzési és kedvezményezési módszer esetére.
- *Részhálózatok* beágyazott forgalmi vizsgálata.

Mivel a speciális esetek kezelésének egyik fő jellemzője az, hogy a hálózat egyes részeinek, vagy pontosabban szólva a hálózati pontok és/vagy élek halmazai egyes *részhalmazainak* *megjelölését*, kiemelését teszi szükségessé, röviden ki kell térnünk ennek módszereire is. A lehetséges alapeszközök ill. módszerek:

- *Interaktív kijelölés*  
Ha a programrendszer interaktív modul is tartalmaz, vagyis lehetővé teszi valamely könnyen kezelhető *szöveges* és/vagy *grafikus* felületen a felhasználói kapcsolattartást, akkor a hálózatrészt esetenként közvetlenül is kijelölhetjük. Ez természetesen csak akkor célszerű, ha viszonylag kicsi és gyakran változó részhalmazokról van szó.
- *További jellemzők*  
A részhalmaz kijelölésére bevezethetünk a meglévők mellé *további* pont és/vagy éljellemzőket. Ezeket külön adatállományokban tároljuk, az esetenkénti felhasználói feladat a megfelelő jellemző (adatállomány) kiválasztása. Viszonylag nagy és gyakran változó részhalmazok esetén javasolt.
- *Bővített jellemzők*  
A meglévő, egyébként is szükséges jellemzők *azonosítási-kódolási* rendszerét úgy is megtervezhetjük, hogy ezek a fő jelentésük mellett a specialitásokra is utalhassanak.



(Például egy szakasz négy számjegyű útkategória adatában az első két jegy kifejezheti magát az útkategóriát, a másik két számjegyet egy speciális jellemző - pl. a teherforgalom kitiltás - meglétét, vagy hiányát.) Akkor javasolt, ha egy alkalmazásban az illető specialitás viszonylag stabil.

- *Automatikus értelmezés*

Van olyan eset is, amikor a részhálózatot nem szükséges teljesen specifikálnunk, mert részbeni megadása - a hálózat által *automatikusán definiált összefüggésekkel* együtt - elegendő a teljes meghatározottsághoz. (Például ha egy részhálózatba bizonyos élek és végpontjaik tartoznak, akkor elegendő csak az élhalmazt megadni, a ponthalmaz meghatározása automatikus.)

## 4.2. Forgalmi tiltások

A *forgalmi tiltások* olyan általános, szokványos esetei, mint az egyirányú szakaszok, befordulástiltások (mint korábban láthattuk) természetesen megjeleníthetők már az alapmodellben is az éljellemzők és pontjellemzők szintjén. Ezek a tiltások általában forgalomfüggetlenek. Itt azzal a - gyakorlatban gyakran előforduló - esettel foglalkozom, amikor a tiltás forgalomtól, pontosabban *forgalmi rétegtől* függ (pl. a belvárosban a teherforgalom tiltott).

Mint korábban a csomóponti transzformációnál (2.2.1.pont) már láttuk, a modellben a tiltásokat általánosan úgy valósítjuk meg, hogy a transzformált hálózatban már nem szerepelnek a tiltott alaphálózatbeli elemek, illetve transzformált megfelelőik.

Természetesen adódik a *rétegtiltás* ugyanilyen módú kezelése. Ez lehetséges, hiszen a transzformációt lépcsőnként hajtjuk végre, *a lépcső mindig automatikusan meghatározza a forgalmi réteget*. Tehát lehet akár rétegenként, de akár lépcsőnként is specifikálni a tiltott részhálózatot. A csomóponti transzformációval párhuzamosan a transzformált hálózatba már nem visszük át az aktuális lépcsőre tiltott éleket és kapcsolataikat, valamint a tiltott élkapcsolatokat.

## 4.3. Relatív forgalmak

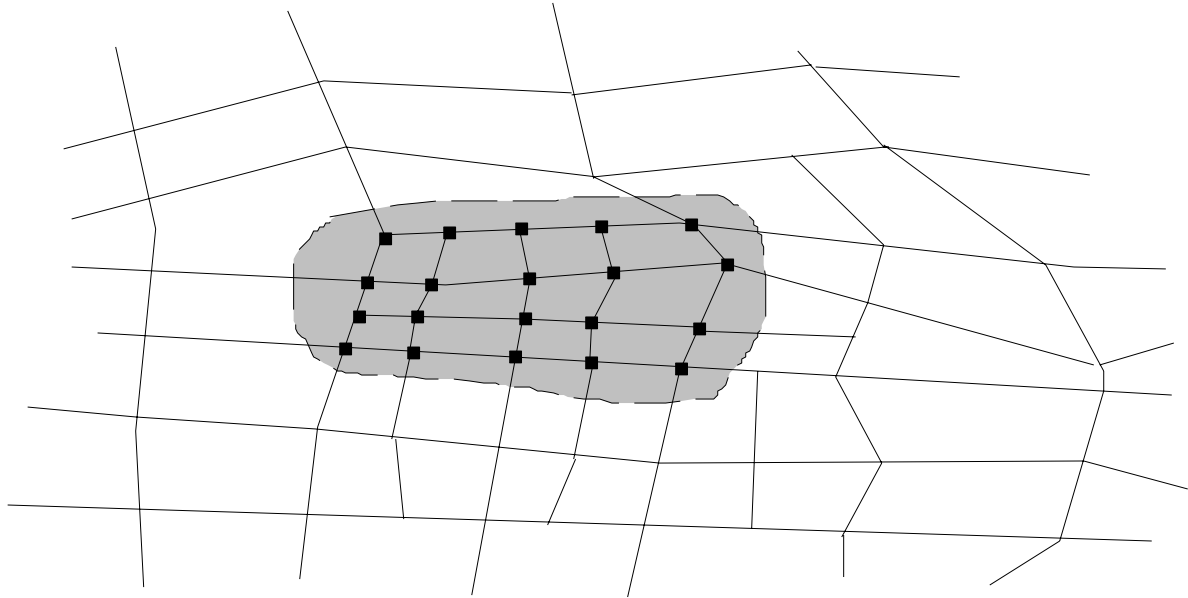
### 4.3.1. Területek

A feladat a teljes tervezési hálózat egy - általában jól lehatárolt, zárt - részhálózatára vonatkozóan számítva, a teljes forgalmi terhelésből *kiemelni* (a valamilyen értelemben) a *részhálózathoz viszonyított*, erre relatív terhelési részt. A részhálózatot nevezzük röviden *területnek*.

A *területre relatív* forgalmi terheléseken értem az alábbi részforgalmakat:

- *Belső forgalom*: a területből (a területbe eső forrás/nyelő pontból) indul, csak belső pontot érint, és célpontja (cél forrás/nyelő pontja) is belső pont.
- *Kiinduló forgalom*: a területből indul, kezdetben belső, majd külső (a területen kívüli) pontokon halad, célpontja külső pont.
- *Cél forgalom*: külső pontból indul, kezdetben külső, majd belső pontokon halad, célpontja belső pont.
- *Átmenő forgalom*: külső pontból indul, célpontja külső pont, de érint belső pontot is.
- *Elkerülő forgalom*: külső pontból indul, célpontja külső pont és nem érint belső pontot.

A fenti felosztás logikailag természetesen nem fedi le a teljes forgalmat, például lehet a területből induló és ugyancsak a területen végződő, de nem csak belső pontot érintő forgalom is. Az ilyen "szabálytalan" forgalmak gyűjtése ugyanúgy lehetséges, egyik elképzelhető hasznosításuk lehet pl. az, hogy meglétük figyelmezteti a közlekedéstervezőt a terület *specifikációjának pontatlanságára*.



22.ábra. Relatív forgalom: terület

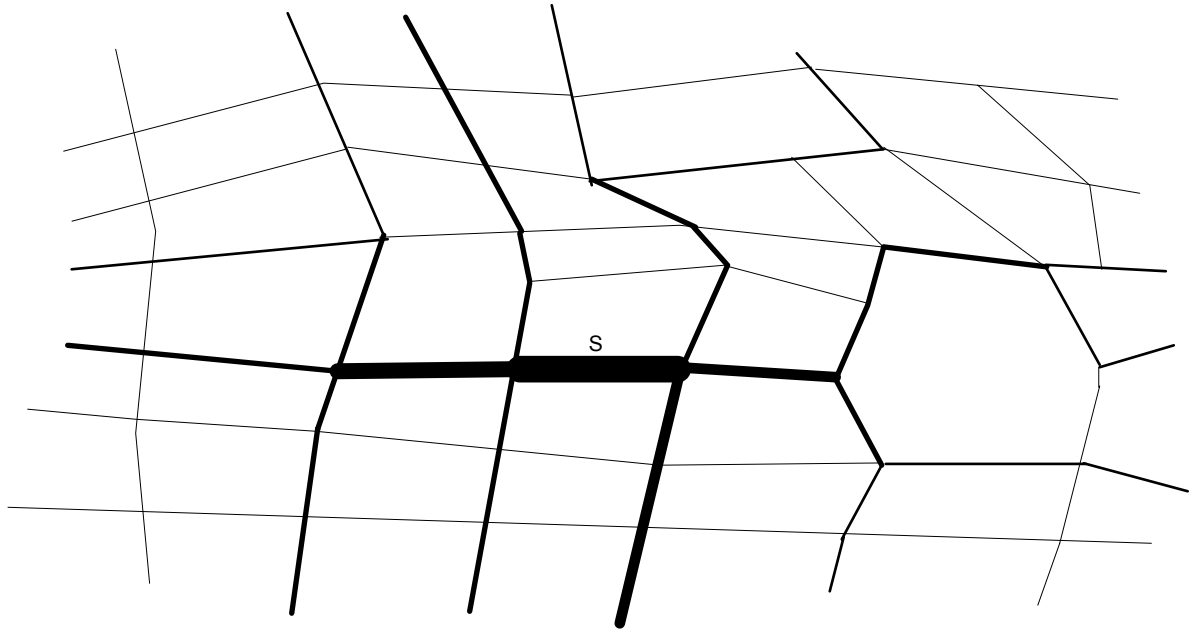
A hálózati modellben elegendő a *területet*, mint *ponthalmazt* - tehát mint a *belső pontok halmazát* - megadni (22. ábra).

A relatív forgalmak gyűjtéséhez - mind az alap, mind a transzformált hálózatban - *új éljellemzőket* kell felvennünk. A gyűjtés (illetve az elkülönítés) csak az *útvonalválasztás-ráterhelés* modulban lehetséges, amikor megvan az információ az *útvonalak* összerakásához és a területhez viszonyított *osztályozásához* (belső, külső stb.). Az aktuálisan ráterhelendő forgalmat, az útvonal minden élére, a globális gyűjtők mellett az ilyen értelemben vett relatív gyűjtőkhöz (éljellemzőkhöz) is hozzáadjuk.

#### 4.3.2. Szakaszok, élek

A feladat a teljes tervezési hálózat egy olyan részhálózatára vonatkozóan megadni az ehhez *viszonyított*, erre relatív terhelési részt, amely részhálózatot egy *élhalmazzal* adunk meg. A leginkább alkalmazott speciális eset az, amikor a halmaz egy él, vagy egy *szakasz* két irányát tartalmazza (23. ábra). Ezt az esetet röviden *pók-nak* (*Verkehrsspinnne*) is nevezik. Az ábrán az S -sel jelölt szakasz két iránya alkotja a részhálózatot, a vastagítások forgalmi terheléseket jelölnek.)

A részhálózatra relatív forgalmi terhelésen értem azt a részforgalmat, amely a *részhalmazba eső élek* közül *legalább egy élen áthalad*. Ez tervezői szempontból hasznos információkat ad egy él, egy szakasz forgalmának összetevőkre bontásához, analizálásához.



23. ábra. Relatív forgalom: pók

A hálózati modellben elegendő a *részhálózatot* mint *élhalmazt* megadni. A hálózati modellbe és a ráterhelés folyamatába való beillesztésre vonatkozóan - értelemszerűen alkalmazva - ugyanaz igaz, mint a terület esetén.

#### 4.4. Útdíjas autópályák

##### 4.4.1. Zárt rendszer

A leginkább elterjedt és legegyszerűbb díjképzési rendszer az un. *zárt* rendszer, amelynél az autópályahálózat minden élére adott illetve egyértelműen számítható egy autópályadíj-költségadat. Ez függ:

- Az él hosszától, általában (de nem feltétlenül) ezzel egyenesen arányos.
- A járműkategóriától (modellünkben ez forgalmi réteggént jeleníthető meg).
- A szakasz útkategóriájától (kiépítettség, szolgáltatási szint)

A modellbe való beillesztésnél a szakasztól való függés egy új alaphálózatbeli *éljellelmezővel* (útdíjas szakasz) jeleníthető meg. Ez egyben meghatározza magát az útdíjas *részhálózatot* is (a jellemzővel rendelkező élek és végpontjaik).

Egy útdíjas él teljes költsége ebben a rendszerben egy utazásra vonatkoztatva általánosan az alábbi képlettel fejezhető ki:

$$g = c + l * b * f$$

ahol  $g$  az él teljes utazási költsége,  $c$  az élnak az autópályadíjtól független költségadata,  $l$  az élhossz,  $b$  a forgalmi réteg által meghatározott alapidj (kilométerdíj), az  $f$  pedig a szakasz útkategóriájától függő korrekciós faktor.

Maga az (autópályadíjból adódó) élköltség rétegfüggő, így lépcsőnként újraszámítandó. Ez a számítás egyszerűen beilleszthető a hálózattranzformációba, az aktuális éellenállás számításának részeként.

#### 4.4.2. Nyílt rendszer

A zárt rendszerrel ritkábban, de egyes alkalmazásoknál találkozhatunk a nyílt rendszerrel is, amelynél az autópálya-részhálózat bizonyos elemein az ún. *kapukon* való áthaladás esetén az utazó egy meghatározott díjat, *kapudíjat* fizet. Ez függ:

- A járműkategóriától.
- Magától a kaputól.

A kaput tartalmazó hálózati elem lehet egy *él*, de egy *pont*, vagy ezen belül akár csak egy *élkapcsolat* (fordulási irány) is.

A modellbe való beillesztésnél az első eset (él-kapu) egy új alaphálózatbeli *éljellelmezővel* jeleníthető meg. Ez megadja rögtön azt az alapértéket is, amely a kapura jellemző és amely mindig az aktuális rétegtől függően módosul. A második eset (pont-kapu) modellbeli megfelelője egy új alaphálózatbeli *csomóponti* jellemző, egy *élkapcsolati típusú mátrix*, amely a kapus csomópontban érvényes (esetleg iránytól is függő), a kapura jellemző alapértékeket adja meg.

Maga a kapudíj rétegfüggő is, így lépcsőnként újraszámítandó. Ez a számítás is egyszerűen beilleszthető a hálózattranzformációba, az aktuális él és/vagy csomóponti éellenállás számításának részeként.

Az egyértelműség kedvéért hangsúlyozom, hogy a fenti *zárt-nyílt* modell megkülönböztetés független a díjbeszedés technikai megvalósításától, az útdíjat többnyire a zárt rendszerben is kapuk segítségével szedik be, a modellbeli eltérés a díjszámítás módjára vonatkozik.

#### 4.4.3. Egyszerű kedvezmények

A kedvezményezési módokat alapvetően a számítástechnikai modellbe való *beillesztés* szempontjából vizsgálom. Ilyen értelemben nevezem *egyszerűnek* azokat a kedvezményezési lehetőségeket, amelyek a ráterhelési alapmodellbe úgy beilleszthetők, hogy a meglévő hálózati és forgalmi elemeket használjuk, tehát *minőségileg új* elem képzésére nincs szükség. Alaprendszerként a *zárt* rendszert tételezzük fel.

- *Területi kedvezmény*  
Az azonos útkategóriájú élek alapdíja a hálózat egyes részein eltérő lehet. A maximumtól való lefelé eltérítést, tehát kedvezményt alkalmazhatunk például a jelenleg relative alacsony átlagos forgalmi terhelésű területeken. Ezeknél az autópályának a magasabb szolgáltatási szintből adódó forgalomvonzó hatása kisebb mint egy közlekedési szempontból frekvenciáltabb hálózatrészen. A területi kedvezmény a forgalomvonzó hatást erősítheti. Ez a kedvezmény egyszerűen modellezhető új útkategóriák bevezetésével. Az új kategóriák a meglévő kategóriák terület szerinti továbbosztásával jönnek létre, így a terület szerinti díjeltérítés az él útkategóriájához tartozó  $f$  korrekciós faktoriall kifejezhető.
- *Időszaki (szezonalis) kedvezmény*

A díjtérítés nyilvánvalóan időszakfüggő is lehet, például idegenforgalmi főszezonban várhatóan magasabb mint azon kívül. Ez egy nagyon egyszerű eset, mivel nem igényli a modell megváltoztatását. A ráterhelési eljárásokban a forgalmi adatok egy viszonylag rövid időszakra, általában egy órára, vagy legfeljebb egy napra vonatkoznak. A modellbeli időszak ennek következtében homogénnek tekinthető.

- *Bérlet jellegű kedvezmények*

Az autópályát potenciálisan rendszeresen igénybevevő forgalom, például a munkabajárás díjkedvezményben való részesítése több szempontból is célszerű. Erre egy lehetőség az adott időszakra vonatkozó bérlet, amelynek ára alacsonyabb mint a várható mértékű rendszeres igénybevétel díja. A ráterhelési modellben a bérletes forgalom a meglévő forgalmi rétegek továbbosztásával mint egy új forgalmi réteg jelenik meg. Ennek  $b$  alapdíja az eredeti alapdíjnak az arányosan csökkentett hányada. Az arányossági tényező a várható mértékű rendszeres igénybevétel összdíjának és a bérlet árának hányadosa.

#### 4.4.4. Viszonylati kedvezmény

A közlekedéspolitikailag is fontos *autópálya díjkedvezmény* rendszer egyik lehetséges eleme a *viszonylati kedvezmény*, amikor az autópálya-részhálózat meghatározott be- és kilépési pontjai között a pályán haladó jármű útdíja kisebb, mint ami az általa a be- és kilépési pont között - a normál díjszabás szerint számítva a legkedvezőbb - megtett út normál útdíja lenne. A kedvezmény járműkategóriánként is változhat.

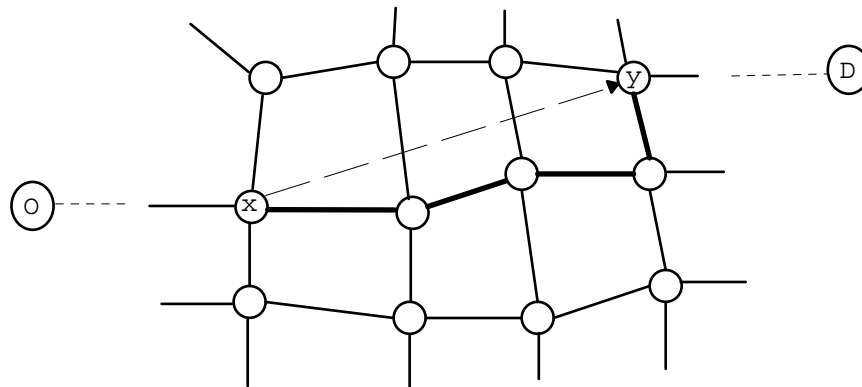
A *viszonylati kedvezmény* a modellezés szempontjából az előzőekhez képest egy *bonyolultabb* elem, aminek elsősorban az az oka, hogy egy él költsége függ attól is, hogy az él *melyik út részéként* veendő figyelembe. Ez másképpen úgy is fogalmazható, hogy egy *út költsége nem feltétlenül egyenlő* az útba eső hálózati elemek (élek, pontok) költségeinek összegével. A probléma egy lehetséges megoldása ([50]) a *viszonylati él* és a *viszonylati út* bevezetése a modellbe.

Az útdíjas autópálya-részhálózat általában a teljes tervezési hálózat nem egy összefüggő része, hanem több önmagában összefüggő részhálózatból áll. Feladatunk szempontjából elegendő ezeket a részhálózatokat külön-külön vizsgálni, más szóval feltételezzük, hogy *a kedvezményezett viszonylat be- és kilépési pontja ugyanabban az összefüggő útdíjas részhálózatban van, valamint a kedvezményezett jármű a belépéstől a kilépésig folyamatosan a részhálózatban halad.* (Elvben elképzelhető és modellezhető lenne a "kombinált" eset is, amikor a kedvezmény több egymással nem feltétlenül összefüggő viszonylat végigutazásához kötődik, de ennek figyelembevétele nagy valószínűséggel csak feleslegesen bonyolítaná a modellt.)

Tételezzük fel még, hogy a *normál útdíj* - az egyéb költségtényezőktől függetlenül - *szakaszonként számítható*, tehát az *alaprendszer zárt rendszer*, valamint a *viszonylati kedvezményes útdíj* (a részhálózaton belül) *útvonaltól független* (nem kapcsolódik hozzá útvonal előírás) és a *kedvezmény valós*, vagyis az adott viszonylatban nincs olyan útvonal, amelyen haladva a normál szakaszdíjak összege a kedvezményes útdíjnál kisebb lenne.

A viszonylati kedvezmény nyújtásának illetve a kedvezményes díj beszedésének a technikai lebonyolítása természetesen többféle is lehet. Itt azt a lehetőséget modellezem, amely legalábbis elméletben úgy írható le, hogy az utazó a normál szakaszdíjakat nem fizeti, hanem a kilépéskor egy meghatározott, csak a viszonylattól függő összeget fizet. Ez az összeg a *viszonylati útdíj*, lehet akár nulla is.

A hálózat  $O$  forrás és  $D$  nyelőpontja között (24a. ábra) haladó jármű költségminimumra törekvő útvonalválasztását nyilvánvalóan befolyásolja az, hogy ha valamely  $x - y$  viszonylatban kedvezményt kap. Az *útvonalkereső algoritmusok* számára a hálózati modellben meg kell adni ezt a választási lehetőséget. Az  $x - y$  viszonylati kedvezményt csak az kapja, aki az  $x$ -ben lép be és az  $y$ -ban lép ki. A gráfban a kívánt útvonal egy közvetlen  $(x, y)$  él felvételével nyilván létrejön. Nevezzük ezt a továbbiakban *viszonylati élnek* (24a. ábra, szaggatott vonal). Külön is megjegyzem, hogy a viszonylati él a modellben egyirányú, csak  $x - y$  irányban járható, ha a kedvezmény a másik irányban is fennáll, azt ettől függetlenül, egy másik viszonylatként kell kezelni.



24a. ábra. Viszonylati él és út

A viszonylati éllel látszólag elválasztottuk a kedvezményezett forgalmat az egyéb forgalomtól. Ez az elválasztás természetesen még a modellen belül korrigálandó, hiszen költségfüggvényeink forgalomfüggőek, és a viszonylati élen haladó jármű ténylegesen a hálózat valós elemein, szakaszain és csomópontjain halad, és befolyásolva ezek forgalmát tehát költségét is. Tehát a viszonylati él behozatala a hálózati modellbe két megoldandó problémát vet fel:

- a) a viszonylati él jellemzőinek megállapítása az útvonalválasztáshoz ,
- b) a viszonylati élre osztott forgalom valós hálózati elemekhez rendelése.

A viszonylati él teljes  $g$  utazási költsége, egy utazásra vonatkoztatva az alapszisztemnek megfelelően:

$$g = c + k$$

alakú, ahol  $c$  az autópályadíjtól független költség, a  $k$  pedig a viszonylati díj.

A viszonylati élt úgy kell paraméterezni, hogy a vele létrejött  $O - \dots - x - y - \dots - D$  útvonal választása a hálózatban reálisan is *teljesíthető* legyen. tehát legyen a hálózatban olyan valódi út, az  $O - \dots - x - \dots - y - \dots - D$  típusú utak között, melynek ugyanakkora a teljes utazási költsége. Mivel az útvonalválasztás költségminimumra törekszik és a minimális költségű út minden része is rendelkezik a minimumtulajdonsággal, elegendő az  $x$  kezdőpontú és  $y$  végpontú utakat vizsgálni.

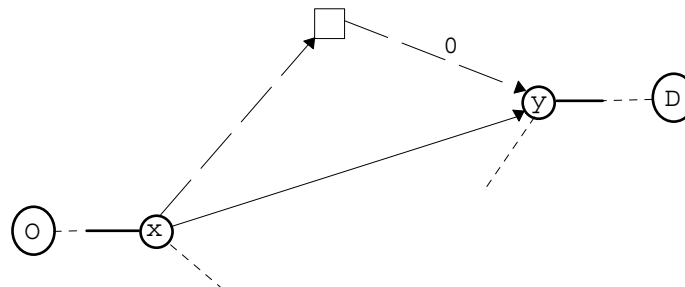
Mivel a  $k$  viszonylati díj konstans, a viszonylati kedvezményt választó utazó az autópályadíjtól független költségösszetevőjét minimalizálja. Tehát ez az utazó az  $x - y$  viszonylatban az autópályadíjtól függetlenül, csak a többi költség tényezővel számított minimális költségű úton halad. Ezt az utat nevezzük viszonylati útnak (a 24a. ábrán a vastag vonallal jelölt út) és ennek az autópályadíjtól független költsége lesz a viszonylati él  $c$  költségösszetevője. Ezzel a viszonylati él hálózatba illesztése teljes, mindkét fenti probléma megoldott az alábbi jellemzőkkel:

- A viszonylati él egyirányú, a belépési pontból a kilépési pontba mutat.
- A viszonylati élre a kezdőpontban minden a kezdőpontba befutó élről rá lehet fordulni.
- A viszonylati élről a végpontban minden a végpontból kiinduló élre rá lehet fordulni.
- A viszonylati él költsége a viszonylati út útdíjmentesen számított költségének és a viszonylati díjnak az összege.
- A viszonylati élre osztott forgalmat a valós hálózaton a viszonylati úthoz kell hozzárendelni, ezzel megnövelve az út egyéb terhelésekből adódó forgalmát.

A fentiekből az is következik, hogy a viszonylati díj csak akkor jelent valódi kedvezményt, ha kisebb mint a viszonylati út normál útdíja.

A viszonylati út a lépcsős ráterhelés folyamatában változhat, áthelyeződhet, hiszen a költség forgalomfüggő, tehát a viszonylati élt és utat is lépcsőnként újra kell számolni.

A viszonylati díj természetesen rétegfüggő is lehet. Ez a lépcsős ráterhelésnél nem jár újabb modellezési problémával, hiszen egy lépcsőn belül csak egy réteghez tartozó forgalom van.



24b. ábra. Viszonylati él fiktív ponttal

Megjegyzendő még, hogy abban az esetben, ha a hálózatban egy valódi  $(x, y)$  él is van, akkor egy technikailag szükséges fiktív pont és él hozzáadásával lehet csak a viszonylati élt beilleszteni (24b. ábra). Az ábrán a négyzet a fiktív pontot jelöli, a bele mutató él a viszonylati él, a belőle induló fiktív él él költsége mindig nulla. A viszonylati út az  $x - y$  közvetlen úttól eltérő is lehet.

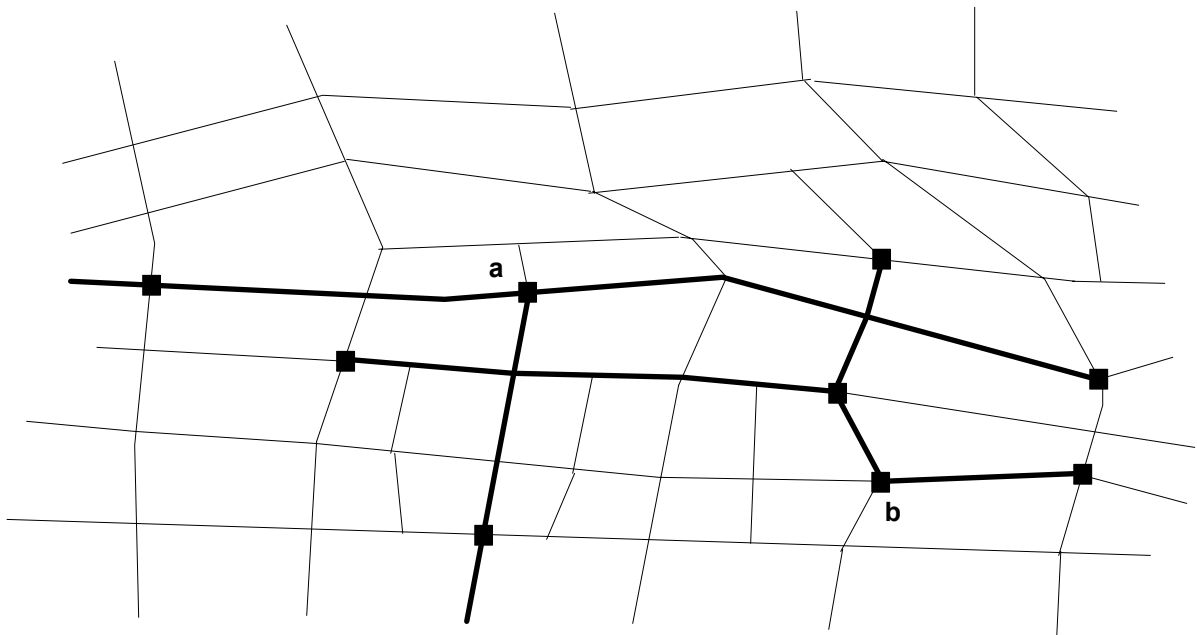
#### 4.5. Részhálózatok

Az eddigi alkalmazások során az autópályákkal kapcsolatban fordult elő, de más esetekben is felmerülhet tervezői igény bizonyos hálózatrészek olyan jellegű részletesebb vizsgálatára, amely a részhálózat bizonyos pontjai közt lebonyolódó forgalmakat emeli ki. Pontosabban

fogalmazva: ha az **a** és **b** a részhálózat egy-egy megjelölt pontja, akkor keressük azt a forgalmat, amely:

- a részhálózatba az **a** pontban lép be,
- a részhálózatból a **b** pontban lép ki,
- az **a** - **b** viszonylatot tekintve végig a részhálózatban halad

valamint az ehhez a viszonylathoz, illetve forgalomhoz tartozó további értékeket (utazási hossz, idő, költség stb.). Az összes megjelölt pontot tekintve ezek az adatok egy-egy *mátrixot* alkotnak (forgalmi mátrix, hosszmátrix stb.).



25. ábra: Részhálózat

A hálózati modellben a hálózatrész- egy éljellemzővel - mint *élhalmaz* emelhető ki, tehát a *részhálózatot* a *kiemelt élek* és *végpontjaik* alkotják. A külön is megjelölt, tehát a mátrixba bevonandó pontok egy *ponthalmazzal* specifikálhatók (25. ábra).

Az eredményadatok gyűjtése (illetve az elkülönítése) csak az *útvonalválasztás-ráterhelés* modulban lehetséges, amikor megvan az információ az *útvonalak* összerakásához, az útvonal és a részhálózat viszonyának vizsgálatához. Emiatt - mint a részhálózatához tartozó adatstruktúrákat - ténylegesen fel kell vennünk a megfelelő *gyűjtőmátrixokat*. Természetesen ugyanígy lehetséges a *rétegenkénti* eredmények képzése.



## 5. A NETWINFO információs és tervezőrendszer

### 5.1. Bevezetés

Ez a fejezet egy konkrét megvalósítást ír le. A dolgozatban tárgyalt elvek és módszerek alkalmazásaként készült el egy több évig fejlesztett, általam tervezett, nagy méretű és komplexitású számítógépes programrendszer, amely - gyakorlati tervezési feladatokhoz - széleskörű hazai alkalmazásra került és néhány külföldi felhasználási referenciával is rendelkezik, valamint a Széchenyi István Főiskolán szakirányú oktatási anyagként is szolgál. Ennek felépítését, főbb paramétereit és szolgáltatásait ismertetem ebben a fejezetben. Természetesen ez az ismertetés más szempontok szerint súlyoz mint egy programdokumentáció vagy egy felhasználói kézikönyv. A rendszer mint szoftvertermék leírását az [54] dokumentáció tartalmazza.

A NETWINFO egy közúti *információs és közlekedéstervezési* programcsomag, és az alkalmazások során a körülötte kialakult tervezési-alkalmazási tudásanyag és *metodika*. A programrendszer magja a lépcsős forgalomráterhelési eljárás, de ez körül van véve *információs, adatelőkészítő és eredményfeldolgozó* modulokkal, ide értve nem csak a technikai modulokat, hanem olyan, önmagukban is összetett és *önállóan is alkalmazható* részeket is, mint a *kalibrációs* modul, amely körzetszintű forgalmi mátrixokat állít elő keresztmetszeti számlálási és más jellemző adatok felhasználásával. A funkciók széles köre *grafikusan* is támogatott.



26. ábra. Főmenü kép

A programcsomag immár egy több mint másfél évtizeddel ezelőtt elkezdett fejlesztőmunka eredménye, pontosabban - mivel fejlesztése jelenleg is folyik - egy közbeeső állomása. A fejlesztés és alkalmazás egyes állomásairól beszámoltunk hazai [7], [8], [9], [30], [55] és külföldi [29], [45] szakmai konferenciákon, valamint szakfolyóiratokban [44], [46], [47], [48], [50], [51], [53],[5], [6]. Az *alkalmazási referenciákat* a függelék tartalmazza.

A fejlesztés annak idején a győri Közlekedési és Távközlési Műszaki Főiskolán indult, később a fejlesztések és alkalmazások súlypontja áthelyeződött a győri BAUCONSULT Mérnökiroda Kft. -hez. Jelenleg is ez a cég a rendszer fő referencialhelye.

Néhány szűkebb értelemben vett *számítástechnikai* információ:

- A programrendszer első (még nem interaktív) változatai Fortran és PL/I nyelven íródtak és a megfelelő ESZR gépeken (R20-R22-R40) futottak. A nyolcvanas évek második felében váltottunk át az akkor erre a célra már egyre alkalmasabbnak látszó személyi számítógépes (IBM PC AT) bázisra, részben az *interaktivitás*, és ezen belül az interaktív grafika könnyebb megvalósíthatósága, részben a *gépi háttér* olcsóbb és egyszerűbb elérhetősége okán. Mivel a feladat erősen belső *adatszerkezet* és *algoritmusingényes* a Turbo Pascal fejlesztő rendszert választottuk. Ez a választás szerencsésen visszaigazolódott, amikor a kilencvenes évek elején megjelent a Borland Pascal 7.0 fejlesztőrendszer, a 16 bites védett (*protected*) programmóddal, számunkra ebből legfőbb előnyként szállítva a memória programból való egyszerű (max. 16MB heap) elérését, ezzel lehetővé téve, hogy a bonyolult, időigényes ráterhelési algoritmusok az operatív tárban, belső adatszerkezeteken fussanak. Így egy nagyobb (4-5000 alaphálózati pont) hálózat egy-egy variánsának teljes ráterhelése is lefut elfogadható válaszidő alatt, lehetővé téve több hálózati és forgalmi változat elemzését.
- A program jelenlegi verziója DOS környezetben fut. *Tervezett* a Windows Delphi (32 bites) platformú átírás, ami két szempontból is javítana a rendszeren. Az egyik a grafikus szolgáltatások még bővebb köre és méginkább könnyen kezelhetősége, a másik az, hogy a 16 bites környezet a nagyméretű alkalmazásoknál (ezen belül is a kalibrációnál) már érezhetően lassítja a rendszert, csökkenti a válaszidőt. A jelenlegi szoftver környezetben (a programcsomag DOS protected módú alkalmazás) a minimális operatív tárnagyság 16MB, az optimális 48MB, úgy hogy ebből 32MB, mint RAM Disk használatos.

* 21.09.1997	Kriebernegg - Kaufmann	Bauconsult	12:01:02	*
-	Közlekedési hálózatok tervezése és vizsgálata			-
Globális_adatok	GYORDET		Globális_adatok	
Az adatbázis teljes neve.....: GYÖR				
Az útkategória adatállomány neve...: Uárosi				
Utkategóriák száma.....: 27				
A hálózati modell neve.....: GYÖR RÉSZLETES				
Pontok száma.....: 1851				
Szakaszok száma....: 2602				
Szakaszok összhossza.....: 529.719 km				
Hálózat összefüggősége.....: Igen				
Utak száma.....: 821				
Utak összefüggősége.....: Nem				
Település adatállomány.....:				
Települések száma.....: 0				
- 1 -				
Home End ↑↔ PgUp PgDn Enter Esc				

27. ábra. Munkalap/1

- A jelenlegi (1998 év eleji) verzió a 9.40 verziószámot viseli. A megoldható feladat nagyságát illetően - feltételezve a ma már általánosnak tekinthető min. 16MB RAM számítógépkonfigurációt - alapvetően csak a szoftverkörnyezetből adódnak korlátok. A programcsomag által kezelhető hálózat maximum 30000 pontot és 180000 élt (90000

szakaszt) tartalmazhat. A körzetek és a forrás/nyelő pontok száma a ráterhelési feladatnál ezen belül nem korlátozott. A kalibrációs modul esetén (a reális, maximum néhány órás programfutási időn belül maradás kedvéért) a forrás/nyelő pontok száma max. 1500 lehet.

- A programrendszer sok és sokféle szolgáltatást ad, ennek megfelelően bonyolult és (személyi számítógépes mércével mérve) nagyméretű. Egy jellemző adat: a Pascal forrásnyelvű *programszövegek* összterjedelme közel 50000 sor. A csoportmunkában történő és állandóan folyó fejlesztés és karbantartás igényeinek megfelelően szigorúan a *modularitás* és *strukturáltság* követelményeit szem előtt tartva tervezünk és programozunk. A programrendszer (jelen verziójában) 70 programot tartalmaz, amelyeket egy keretprogram fog össze.
- Az *adatkezelést* - a programrendszer részeként kizárólag e célra kifejlesztett és emiatt eléggé hatékony - *saját adatbáziskezelővel* oldjuk meg, ahol ez szükséges, biztosítva a *grafikus* adatfelvitel és *karbantartás* lehetőségét. Szintén belső komponens a nagyobb tömegű adatrögzítés célját szolgáló, bizonyos előellenőrzéseket is megvalósító speciális *szövegszerkesztő*. A más rendszerekkel való adatcsere, *import* és *export*, valamint a rendszertől független adatmegőrzés, *archiválás* és a tömör adattovábbítás céljára a mindenhol ismert *szövegfájl* formátumot használjuk.

* 23.09.1997		Kriebernegg - Kaufmann			Bauconsult	21:10:20		*	
-		Közlekedési hálózatok tervezése és vizsgálata						-	
Hálózati információ		GYORDET				Hálózati információ			
Index 1851/17	Kezdőpont 00019	X -151817	Y 58639	ULZ 0	Tipus ZT0	Éldb. 4			
Ind	Ut	Kezdőszelv.	Végpont	Végpszelv.	Hossz	OD	Kat	Tipus	Kapu
4/1	SASHEGY	0+000	00021	0+547	547	0	510	1 1	0
4/2	81	79+207	08018	79+857	650	0	170	1 1	0
4/3	81	79+207	08248	76+410	2797	0	170	1 1	0
4/4	KAKASHEGYI	0+000	11501	2+166	2166	0	111	1 1	0
		Körzet: 0080105		75.0%					
↑ ↓ PgUp PgDn Home End \$ B Enter * Esc									

28. ábra. Munkalap/2

- A *grafikus funkciókat* a programrendszeren belül a Pascal képernyőgrafikával oldjuk meg. Az eredmények grafikus megjelenítésénél ez nem mindig elegendő. Jogos felhasználói igény a többletű, tetszés szerint nagyítható, forgatható, feliratozható, *grafikusan továbbfeldolgozható*, papíron is rögzíthető (plotter) eredmény. Ezen követelmények mindegyikét a Pascal grafikus rendszere nem tudja teljesíteni. Itt azt a megoldást választottuk, hogy, a grafikus információt programunk egy, a grafikus rendszerek közt standard módon ismert formátumú fájlok formájában is szolgáltatja, ezek aztán megfelelő grafikus rendszerrel (pl. AUTOCAD) tetszés szerint továbbfeldolgozhatók.
- A programrendszer - induláskor választhatóan - *angol, magyar* vagy *német* nyelven képes kommunikálni. A jellegzetes *bejelentkező* és egyben *szöveges főmenü* típusú képet láthatjuk a 26. ábrán. A tipikus *szöveges munkaképernyőket* mutat a 27. és 28. ábra. Tipikus grafikus képeket az egyes szolgáltatások ismertetésénél közlünk. (Az eredeti képek természetesen színesek.)

Közlekedéstervezői szempontból a rendszer három fő modul tartalmaz:

- *Részterület kiemelés.* Egy kordonokkal körülhatárolt tervezési területrészből és egy a teljes tervezési területhez tartozó körzetszintű *forgalmi mátrixból* egy új, önálló *tervezési szituációt* állítunk elő. Az új *hálózat* a lehatárolt részhálózat lesz, az eredeti körzetbeosztásból és a kordoninformációkból keletkezik az új hálózat *körzetbeosztása*, az eredeti mátrix egy speciális transzformációjával kapjuk az új *mátrixot*.
- *Kalibráció.* Adott körzetszintű forgalmi mátrixot *kalibrálunk*, kiegyenlítünk egy hálózaton adott előírt életterhelési (pl. keresztmetszeti számlálási) adatokhoz mint *peremfeltételekhez*. Az eredménymátrix egyik lényeges jellemzője, hogy visszaterhelve teljesíti a peremfeltételeket.
- *Forgalomráterhelés.* Az ismertetett lépcsős ráterhelés egy megvalósítása.

Részletesebben a rendszer szolgáltatásainak ismertetésénél tárgyalom az egyes modulokat.

A NETWINFO rendszer korai verziói tartalmaztak egy viszonylag egyszerű tömegközlekedési modult is. Ez alapvetően az egyéni közlekedési analógiákon (pl. csomópont-megálló, átszállás-csomóponti ellenállás) alapuló modell volt. A későbbi alkalmazási igények -elsősorban az autópályákkal kapcsolatos feladatok - az egyéni közlekedési modell minél részletesebb és pontosabb kidolgozását, bővítését tették szükségessé, ennek következtében a tömegközlekedési modul "kiszorult" a rendszerből.

## 5.2. A rendszer adatbázisa

### 5.2.1. Bevezetés

A rendszer adatbázisa egymástól *független*, önállóan kezelhető, de bizonyos funkciókban egymással kapcsolatba is hozható adatsomagokból áll. Közlekedéstervezési szempontból egy ilyen adatsomagot egy *tervezési szituációnak* tekintünk és nevezünk. (Az adatsomagok számítástechnikailag is elkülönülnek, a rendszer főkönyvtárban a program-alkönyvtár mellett egy-egy adat-alkönyvtárat alkotnak.)

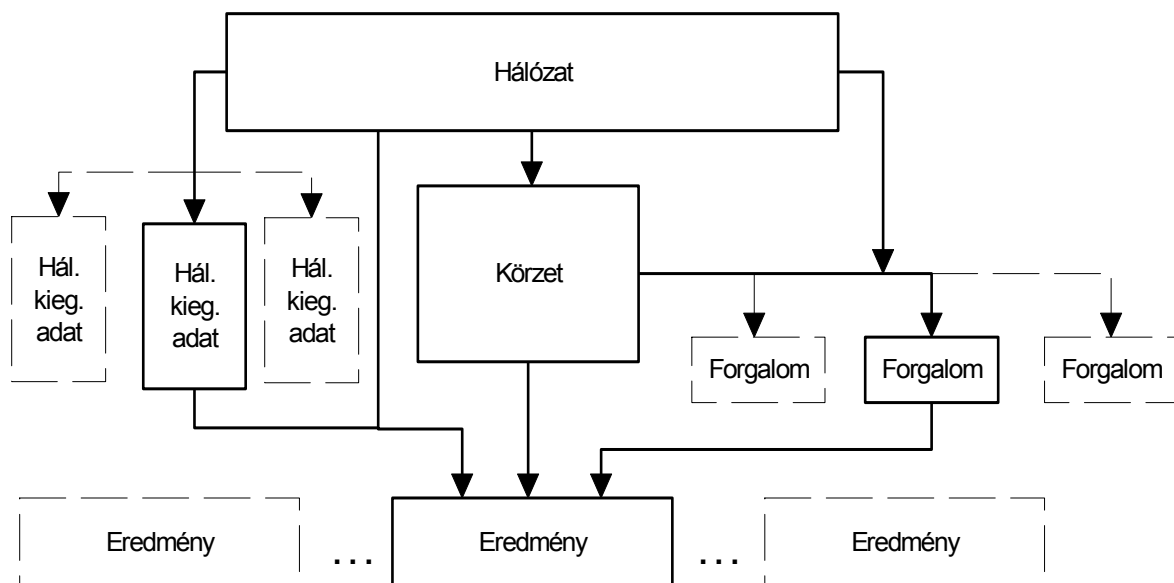
Egy szituációt alapvetően egy tervezési *hálózat* és ennek egy *körzetbeosztása* határoz meg, ezek tehát *egyediek* a szituációban. A szituációhoz több, egymástól független, a hálózattal kompatibilis hálózati *kiegészítő* adatsoport, valamint a hálózattal és körzetbeosztással kompatibilis *forgalmi alapadat* adatsoport tartozhat. A hálózat, a körzet valamint egy-egy kiegészítő és forgalmi alapadatcsoport határoz meg egy *eredmény adatsoportot* a szituációban (ld. 29. ábra, a nyíl iránya a levezetést, a függőséget mutatja).

Egy-egy adatsoportot, a számítástechnikai realizáció szempontjai által megszabott módon, egy vagy több adatállomány reprezentál a szituációban.

Itt a bevezetőben kell még foglalkozni az adatok mértékegységeivel is. A *hosszt* mindig méterben, az *időt* mindig másodpercben, a *költséget* (tekintettel a külföldi alkalmazásokra is) egy közelebről meg nem határozott pénzegységben (G) mérjük, ahol szükséges, törtrészekkel is számolva. A sebesség mértékegysége a belső számításoknál m/sec, a felhasználói szinten km/ó.

A *forgalom* (mint pl. egy mátrixelem, vagy egy szakaszterhelési adat) kétféle dimenzióban is szerepel a rendszerben. Az egyik az elméletben jól ismert és alkalmazott *egységjármű/óra* (röviden E/ó) absztrakt mértékegység, amit az egyes modulok belső számításainál, illetve

néhány eredményadat dimenzionálásánál kell alkalmaznunk, lévén, hogy bizonyos függvények, illetve eredmények csak ebben értelmezettek.



29. ábra. Tervezési szituáció

A másik forgalom-mértékegység az adatok keletkezési módjából (általában számlálás) közvetlenül adódó - valamilyen adott időszakra és/vagy járműfajtára vonatkozó - természetes egység (pl. járműdarabszám). Ezt, mivel tipikusan a körzetszintű mátrixok megadásának módja, rendszerünkben *mátrixegységnek* (röviden *Me*) nevezzük. Az input adatok szinte kizárólag ebben a formában jelennek meg, és az eredményadatok nagy része értelmezhető ilyen mértékegységben is. A rendszer input és output szinten is képes bármilyen forgalmi dimenziót kezelni, mátrixegység esetén természetesen specifikálni kell az E/ó egységre való *átszámítás* faktorait is.

## 5.2.2. Alapadatok

### 5.2.2.1. Hálózatleírás

#### 5.2.2.1.1. Pontok, élek

A hálózat alapvetően *pontokból* és *élekből* áll. Tehát az első rendben megadandó és nélkülözhetetlen adatok ezek jellemzői. A *jellemzők* bizonyos *csoportjai* viszont összevonhatók és az adattárolás tömörsége és nagyobb biztonsága céljából egy-egy kóddal kifejezhetők, helyettesíthetők. Rendszerünkben ilyen összevont *éljellemző* az útkategória adat.

Egy *útkategóriához* az alábbi adatokat rendeljük:

- Kategórianév és kód.
- Megengedett maximális sebesség.
- Elvi forgalmi kapacitás.
- Autópályadíj faktor (zárt rendszer).
- Autópályadíj faktor (nyílt rendszer).

- Üzemeltetési költség faktor.

Az adatokat az *idő* és *költségszámításnál* használjuk fel. A megadott faktorok további - a forgalmi rétegtől és/vagy a konkrét éltől függő - értékekkel együtt képeznek költségadatot. A faktorok itteni feltüntetése gyakorlati okokból célszerű: ha pl. egységesen 10% -kal emelni akarjuk a modellben az autópályadíjakat (vagy az üzemanyag árát), elegendő csak itt, a megfelelő kategóriáknál megváltoztatni a megfelelő faktort.

Rendszerünkben a hálózatléíró adatok elsődlegesen *pontorientáltan* vannak megadva, egy pontnál magához a ponthoz, továbbá a pontból kiinduló élekhez tartozó adatokat találjuk meg.

*Egy él adatai :*

- A végpont azonosító kódja.
- A hossz, és ebből az átkelési hossz aránya.
- Az útkategóriakód.
- Az útkód.
- Engedélyezett irány-e.
- Vízszintes és függőleges vonalvezetési jellemzők.

Az adatokhoz megjegyzem, hogy praktikus adatkarbantartási okokból az aktuálisan egyirányú szakaszok *tiltott iránya* is szerepel (nem engedélyezett irányként), így a megfordítás vagy kétirányúsítás könnyebben átvezethető. A *vonalvezetési jellemzők* az él megtételi idejének számításánál az útkategória által megadott maximális sebességet módosítják. Az útkód jelentését a következő pontban részletezzük.

*Egy pont adatai:*

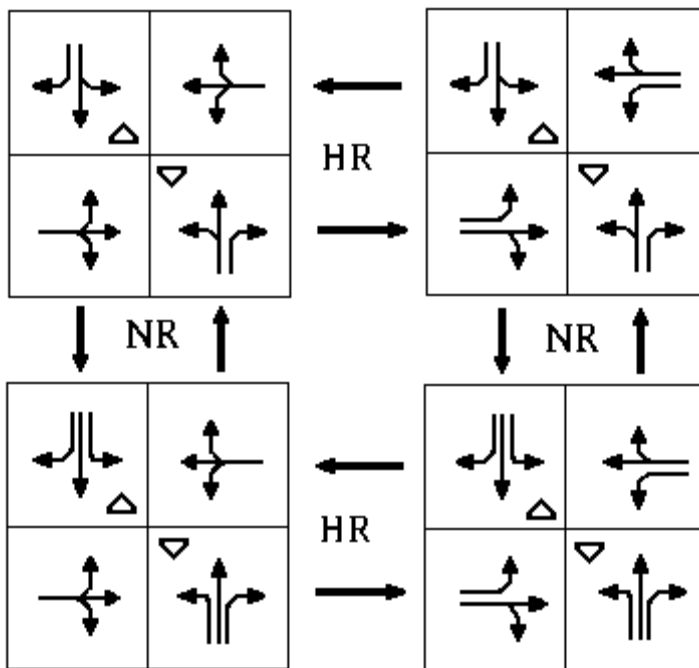
- A pont azonosító kódja és megnevezése.
- A pont koordinátái.
- A kiinduló él számja és élenként az éladatok.
- A pont típusa.
- Forgalmirányító jelzőlámpás csomópontnál:
  - a lámpa fázis (db) száma
  - a lámpa periódusideje
  - a forgalmi irányok fázisokba osztásának kódja
- Az élkapcsolati (befordulási ) jellemzők mátrixa.

Az adatokhoz megjegyzem, hogy *pontazonosításra* egy 8 jegyű alfanumerikus kódot enged meg a rendszer. Ez lehetővé teszi, hogy a felhasználó "beszélő" kódrendszert definiáljon (pl. 2 jegyenként: ország, megye, kisebb területi egység, ezen belül egyedi pontkód). A pontmegnevezés csak demonstrációs célokra szolgálhat, ha megadják. A *koordináták* alapvetően a grafikus megjelenítést szolgálják, bár van automatikus élhossz-számítási lehetőség (koordinátatávolság) is a szolgáltatások között.

A *ponttípus* (és periódusidő) a *befordulási mátrixszal* együtt határozzák meg egy adott forgalomáramlási helyzet mellett a csomópont ellenállását az egyes áthaladási irányok szerint.

A rendszerben alkalmazott *csomóponttípusok*:

- *Konstans* ellenállású csomópontok (általában technikai pl. grafikai okokból felvett pl. nulla ellenállású pontok), az élkapcsolat ellenállása az élkapcsolat jellegétől független.
- *Tábla nélküli* elsőbbségszabályozású (*jobbkezes*) csomópontok
- *Táblás* elsőbbségszabályozású csomópontok: altípusok a *táblák* (elsőbbség tábla, stop tábla) és a *sávelrendezések*, d. pl. 30. ábra. Az ábrán a HR főirányt, az NR mellékirányt jelöl).
- *Forgalomirányító jelzőlámpás* szabályozású csomópontok.



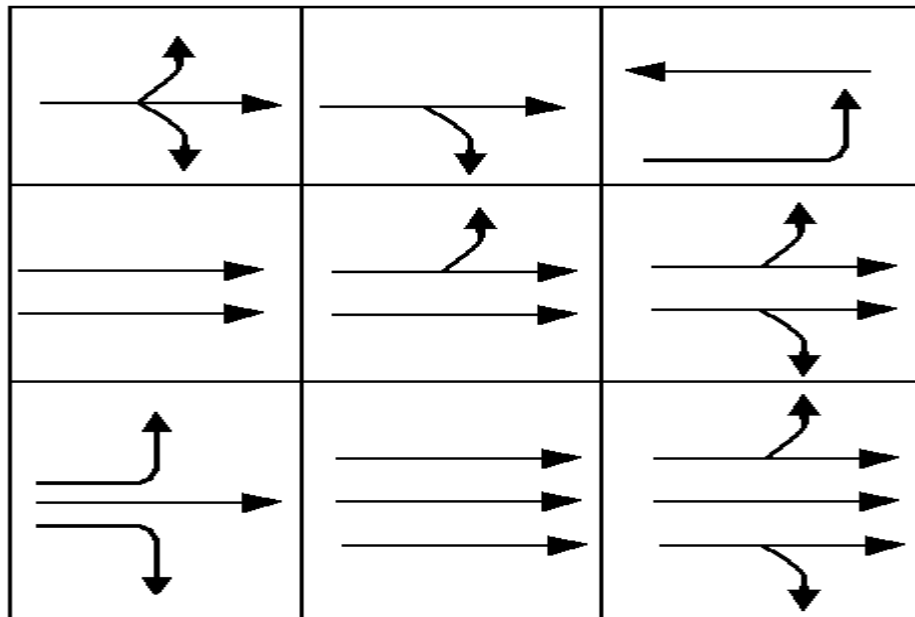
30. ábra. Sávelrendezések

A típusokon belül az *élkapcsolatok jellegét* adjuk meg a befordulási jellemzők mátrixában:

- *A tábla nélküli* csomópontokban megkülönböztetünk *tiltott* irányt, *jobbra kanyarodó* irányt és *egyéb* lehetséges irányt.
- *A táblás* csomópontokban megkülönböztetünk *tiltott* irányt, első és második *főirányt* (30. ábra HR), első és második *mellékirányt* (30. ábra NR), valamint ezeken belül *előre*, *jobbra* és *balra* továbbhaladást.
- *A lámpás* csomópontokban az élkapcsolat jellegét az adott irány *sávképlete* (31. ábra) határozza meg

Megjegyzem, hogy a *haladási irány* (előre, jobbra és balra) elvben a koordinátákból is kiszámítható (ilyen szolgáltatás van is a rendszerben), de ez nem esik mindig egybe a forgalomszabályozás által meghatározott irányminősítéssel, ezért a kézi módosítás, kézi kódolás lehetőségét is meg kell hagyni, tehát az iránykódot is tárolni kell.

A hálózatleírás globális adatai a pontok és élek száma, valamint a hálózat érvényessége. A *hálózatot* akkor tekintjük *érvényesnek* ha összefüggő. Az érvényesség szükséges a *ráterhelés* végrehajthatóságához.



31. ábra. Sávképletek

#### 5.2.2.1.2. Utak, települések

A pontokkal és éllel megadott hálózati struktúrán definiálni kell még egy teljesen más jellegű helyazonosítási rendszert is, a közlekedésben jól ismert és még ma is szinte kizárólagosan alkalmazott *útszám-szelvényt* szerinti azonosítást. A tényleges gyakorlati alkalmazásra szánt tervező rendszerekből ez nem hiányozhat, hiszen - az elméletileg egyszerűbb, pontosabb - és bizonyos rendszerekben be is vezetett *csomópontkód - távolság* szerinti helyazonosítás az eredményeket felhasználó gyakorlati szakemberektől még meglehetősen idegen.

Egy utat az alábbi adatokkal írunk le:

- *Útazonosító* (általában maga az útszám vagy az utcanév).
- *Útkód* (egy rövid belső kód).
- *Az út pontjai és szelvényt* az úton.

A rendszer tartalmazza a beszélynyezés-átszélynyezés lehetőségét is.

Az útrendszert akkor tekintjük *érvényesnek* ha minden út összefüggő, tehát folyamatosan beszélynyezhető. Az érvényesség nem szükséges a *ráterhelés* végrehajthatóságához, csak informatív adat.

Elsősorban *demonstrációs* célokból szerepelhet egy tervezési szituációban egy *település* adatbázis is. Egy településhez megadjuk a település *nevét*, típusát, a település területére eső hálózati *pontokat*, ezen belül a közlekedési szempontból *centrális* pontot. Esetleg demonstrációs okokból célszerű lehet felvenni olyan kisebb településeket is, amelyekhez nincs hálózati pont. Ilyen esetekben egy közel eső pontot adunk meg centrális pontként, de külön megadva a valódi centrum ettől való távolságát és irányát is.



### 5.2.2.1.3. Kiegészítő adatok

Rendszerünkben a speciális feladatok meghatározásához és megoldásához az alábbi típusú hálózati kiegészítő adatokat használjuk:

- *Területleíró* ponthalmaz: pontok felsorolása. A ráterhelés *relatív forgalom* részfeladatánál és a *részterület kiemelés* feladatcsoportnál használjuk.
- *Keresztmetszet* élhalmaz: élek felsorolása a kezdő és végpontokkal, esetleges kiegészítő adattal. A kiegészítő adat szerinti altípusok:
  - *Nincs kiegészítő adat*: a ráterhelés *relatív forgalom* részfeladatánál és a *részterület kiemelés* feladatcsoportnál használjuk.
  - *Forgalomnagyság*: a kalibráció feladatcsoportnál használjuk.
  - *Pontorszám*: részhálózatok beágyazott forgalmi vizsgálatánál használjuk.
  - *Kapudíjfaktor*: a nyílt rendszerű autópálya díjszámításnál használjuk.
- *Kapuleíró* ponthalmaz: pontok felsorolása az irányonkénti kapudíjfaktorokkal. A nyílt rendszerű autópálya díjszámításnál használjuk.

### 5.2.2.2. Körzeteírás

Egy körzet egy összefüggő területrésznek felel meg a hálózatban, mint csomópontok egy halmaza is tekinthető. A körzetnek a rendszerben betöltött funkciójához azonban nem szükséges minden pont, elegendő a *forrás/nyelő* pontokat megadni, a többi pontnak csak demonstrációs jelentése van. A körzeteíró pontok közül egyet - szintén csak demonstrációs célból - *centrális* körzetpontként emelhetünk ki.

Minden körzetben minden *forrás/nyelő* ponthoz meg kell adnunk a pont relatív *súlyát* a körzetben. Egy ilyen súlykiosztást egy *súlyrendszernek* nevezünk. Mivel a pont súlya eltérhet forgalmi rétegenként, több súlyrendszert adunk meg. (Amelyek közül a ráterhelés minden lépcsőjében választhatunk egyet.)

A körzeteírás *globális* adatai:

- *A körzetek száma* és a rendszer érvényessége. A körzetrendszert akkor tekintjük *érvényesnek* ha van legalább két körzet, legalább egy súlyrendszer és minden körzetben van legalább egy *forrás/nyelő* pont. Az érvényesség szükséges a ráterhelés végrehajthatóságához.
- *A súlyrendszerek száma* és a súlyrendszerek azonosítói.

A körzeteírás *körzetenkénti* adatai:

- *A körzet rövid kódja* és teljes megnevezése.
- *A forrás/nyelő* pontok felsorolása, pontonként és súlyrendszerenként a relatív súlyok, százalékban kifejezve.
- *A körzet egyéb* pontjainak felsorolása.
- *A centrális* pont.

*Rendszerünkben a körzetbeosztás diszjunkt, vagyis egy pont legfeljebb egy körzetbe tartozhat.*

### 5.2.2.3. Forgalomleírás

Rendszerünkben a forgalomleíró adatok *forgalmi rétegenként* adottak. Egy forgalmi rétegen (röviden: *réteg*) értjük a forgalom egy viszonylag homogén, egységes módon paraméterezhető szeletét. A réteg leíró adatai:

- globális rétegjellemzők
  - forgalmi mátrixok
- *Globális rétegjellemzők*
    - *Időfaktor*: technikai adat, annak az időintervallumnak a hossza, amelyre a forgalmi mátrixok vonatkoznak. Az  $M_e$  és az  $E/\bar{o}$  közti átszámítás egyik faktora.
    - *Egységjárműfaktor*: technikai adat, a réteg egységjármű egyenértéke. Az  $M_e$  és az  $E/\bar{o}$  közti átszámítás másik faktora.
    - Autópálya *kilométerdíj*: alapadat zárt rendszerre.
    - Autópálya *kilométerdíj kedvezmény*: alapadat zárt rendszerre, a költség számításnál levonható, külön gyűjtése tervezési szempontból célszerű lehet.
    - Autópálya *kapudíj*: alapadat nyílt rendszerre.
    - Autópálya *kapudíj kedvezmény*: alapadat nyílt rendszerre, szerepe és kezelése a kilométerdíj kedvezményével azonos.
    - *Üzemeltetési költség*: fajlagos, 1 km -re vetített üzemeltetési költség, (üzemanyag, karbantartás stb. összevont értéke).
    - *Idő egyenérték*: fajlagos, 1 órára vetített költség, az időmegtakarítás értéke.
  - *Forgalmi mátrixok*

A *forgalmi mátrixok* körzetmátrixok, amelyek az elosztandó forgalmat tartalmazzák a kiinduló körzet-célkörzet viszonylatokban. Megadásuk  $M_e$  egységben történik.

A globális rétegjellemzők az idő és költség számítás faktorai, tehát a megfelelő ellenállásfüggvények paraméterei. Az azonos globális rétegjellemzőkkel bíró forgalmi mátrixok *csoportmátrixba* vonhatók össze. A ráterhelésnél csoportmátrixokat adunk meg input adatként.

## 5.2.3. Az adatok tárolása

### 5.2.3.1. Belső tárolás

A lépcsős forgalomráterhelési eljárásnál a tárkihasználás és számítási idő szempontjából döntő jelentőségű az *útvonalválasztás és ráterhelés* modul. Az ebben kezelt adatok nagyobb - és tárigeny szempontjából meghatározó jelentőségű - része a hálózati elemekhez kötődik, tehát pont vagy éljellemző. A korábban (ld. 2.3.1. pontban) kifejtett általános vizsgálatokból adódó következtetéseknek megfelelően a hálózati adatok tárolására az *éltárolás* módszerét alkalmazzuk, a hardver-szoftver környezet kötöttségeiből adódó módosításokkal és általánosításokkal.

Az alkalmazott programfejlesztő rendszerben egy tömb által maximálisan elfoglalható hely a tárban 65520 bájttal. Az éljellemzők között a legnagyobb tárigenyűek (pl. aktuális élterhelés) 4 bájttal tárolhatók. Következésképpen a tiszta éltárolás alkalmazása az élek számát a 16380 maximális értékben korlátozná.

Ez nagyobb hálózatoknál már az alaphálózati szinten is kevés (kb. 5000 pontnak felel meg), de a transzformált hálózati szinten méginkább erős korlátozás (alaphálózati szintre visszavetítve kb. 5000 él és 1500 pont lenne a felső korlát).

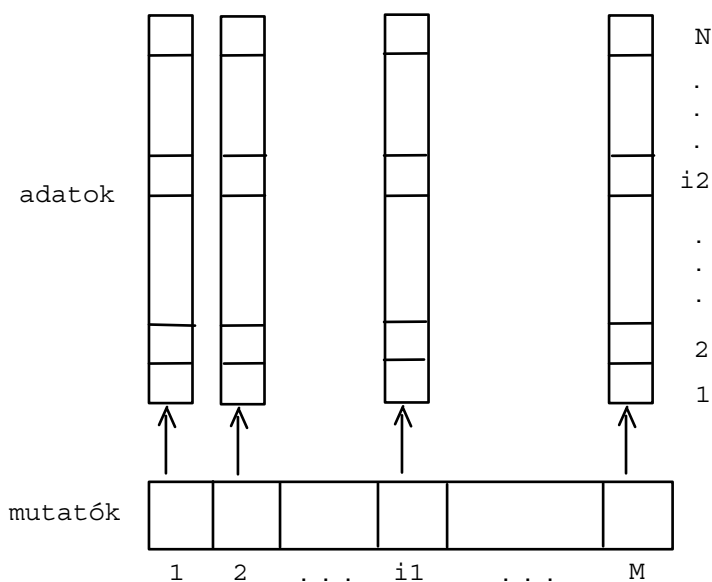
Ennek feloldására az éltárolás tömbjeit összefüggő szakaszokra tördeljük és az egyes darabok közti kapcsolatot egy erre a célra felvett mutatótömbbel biztosítjuk, (32. ábra) ezt az adatszerkezetet szokás *kollekciónak* is nevezni.

A kollekció nem veszi el a tömb lényeges előnyeit, ugyanis a *közvetlen* hozzáférhetőség, az indexelés megmarad, csupán egy egyszerű indexátszámítás szükséges. Jelölje  $i$  az eredeti indexet. Az eredeti tömbben  $i$  indexű elem a kollekcióban az alábbi módon számítható  $i_1$  és  $i_2$  indexekkel érhető el (a 32. ábra jelöléseivel):

$$\begin{aligned} i_1 &:= i \operatorname{div} N + 1 && \text{ha } i_1 > M \text{ akkor } i_1 := M \\ i_2 &:= i \operatorname{mod} N && \text{ha } i_2 = 0 \text{ akkor } i_2 := N \end{aligned}$$

Ezzel a módszerrel a kezelhető hálózat méretét csak az operatív tár nagysága korlátozza. Pl. egy 10000 pontot és 35000 élt tartalmazó alaphálózat és az ebből keletkező kb. 35000 pontos és 100000 éles transzformált hálózat esetén a tárigény kb. 6 Mbájt.

A hálózati elemekhez közvetlenül nem kötődő adatok egyik csoportja (pl. rétegjellemzők, körzetadatok) olyan, hogy méretük, tárigényük kicsi a hálózathoz képest, tárolásuk megfelelően választott egyszerű tömbökben lehetséges és célszerű.



32. ábra. Kollekciónak

A másik csoport (pl. forgalmi mátrixok) olyan, hogy ugyan nagyméretűek, de az operatív tárból való tartásuk magában a ráterhelési eljárásban nem szükséges, az eljárás gyorsaságát gyakorlatilag nem csökkenti, ha egyszerre csak egy viszonylag kis részt (pl. egy mátrixsört) tartunk bent. Ha valamilyen funkció gyorsabb lebonyolításához célszerű egy *mátrixot* ténylegesen *mátrixformában* - illetve ezzel az elérhetőség szempontjából gyakorlatilag

ekvivalens módon - az *operatív tárban* tartani, akkor erre a célra is a kollekción használjuk, a mátrix sorokra vagy oszlopokra tördelésével.

### 5.2.3.2. Külső tárolás

A háttértárolón az adatokat fix rekordhosszúságú, a rekordsorszám alapján közvetlenül elérhető Pascal *típusos fájlokban* tároljuk. A hálózati adatok tárolása itt is pontorientált, egy rekord egy pont, és a belőle kiinduló élek adatait tartalmazza. Hasonló szervezésű a többi adatállomány is (egy körzet egy rekord stb.). Az állományok nagy részénél szükség van a rekord valamely mezője (tartalom) szerinti közvetlen elérésre is. Ezt saját tervezésű, az adott fájlhoz, és a feladathoz legjobban illeszkedő *indexfájlokkal* biztosítjuk.

A ráterhelésnél a hálózathoz kötődő elsődleges eredményadatokat, a rétegenkénti pont és élterheléseket (amelyekből az összes többi eredményadat képződik) egy ugyanolyan szervezésű fájlban tároljuk, mint a hálózati alapadatokat, az indexfájl is közös.

A forgalmi mátrixokat, amelyeknél a legtöbb funkcióban nincs szükség elemenkénti közvetlen elérésre, elemenként egy rekordban, egyszerűen sorfolytonos módon tároljuk a háttértárolón. Ezek feldolgozása tipikusan szekvenciális.

A rendszer üzemeltetési biztonságát, a más rendszerekkel való kapcsolattartást és az adatok számítástechnikai hordozhatóságát nagyban elősegíti, hogy a teljes adatrendszer, a megfelelő programfunkcióval bármikor kimenthető *szöveges adatállományokba*. Ezekből a tervezési szituáció (szintén programfunkciókkal) bármikor visszaállítható, valamint ezeken keresztül lehet megvalósítani a más rendszerből való adatátvitel-*import* - és a más rendszerbe való adatátadás-*export* - funkciókat is.

## 5.3. A rendszer szolgáltatásai

### 5.3.1. Alapfunkciók

Az itt ismertetendő funkciók alkotják rendszertervezési-programtervezési, de alkalmazási szempontból is a *rendszer magját*, lényegében ezekből az eljárásokból épülnek fel a lényeges szolgáltatások.

- *Hálózati transzformációk*  
A rendszerben a korábban (2.2.pont) részletesen ismertetett és indokolt eljárást, illetve ennek egyes önálló részeit: a *csomóponti transzformációt*, a *forrás-nyelő pontok beillesztését* és az *eredmények leképezését* valósítjuk meg.
- *Útvonalkeresés*  
Az első minimális utak meghatározására a 3.2.2.2. pontban leírt faépítő eljárást valósítjuk meg, az aktivitás halmaz kupac-rendszerű tárolásával. A második minimális utak keresése a 3.3.2. pontban megadott eljárás szerint működik.
- *Sebesség és időszámítás*

Egy út megtételének ideje az útvonalba eső hálózati elemek - élek és csomóponti áthaladások (élváltások) - idejének összege. Az *élidő* az *élhossz* és az aktuálisan kialakult *élssebesség* hányadosa. A kapacitáskorlát, mint alapelv következménye, hogy az élssebesség az elvi kapacitás felé közeledve csökken. A csökkenés módját elméleti megfontolások és gyakorlati tapasztalatok alapján exponenciális függvénnyel szokás megadni. A

rendszerünkben alkalmazott sebességfüggvények a [70] kézikönyv javaslatai szerinti épülnek fel, illetve a közlekedéstervezés alapvető ismeretanyagához tartoznak (ld.pl. [32]).

$$V_a = V_e * \alpha * \beta \qquad C_a = C_e * \alpha * \beta$$

ahol  $V_e, C_e$  az általános, az útkategóriából adódó megengedett maximális sebesség (km/ó) illetve elvi forgalmi kapacitás (E/ó), a  $0.2 \leq \alpha, \beta \leq 1$  a konkrét szakaszhoz tartozó vízszintes és függőleges vonalvezetési korrekciós tényezők,  $V_a, C_a$  az élre aktualizált maximális sebesség, illetve forgalmi kapacitás adat, a forgalomfüggő tényleges  $V(F)$  sebesség pedig (m/s egységben):

$$V(F) = \frac{V_a}{3.6} - 0.21 * V_a * \frac{F}{3.6 * C_a} \qquad \text{ha } F < 0.3 * C_a$$

$$V(F) = \frac{V_a}{3.6} - 0.21 * V_a * \frac{F}{3.6 * C_a} - e^{2.25 * \left(\frac{F}{C_a}\right)^3} \qquad \text{ha } 0.3 * C_a \leq F < 2 * C_a$$

$$V(F) = 1 \qquad \text{ha } 2 * C_a \leq F$$

A megfelelő  $T(F)$  időérték (sec egységben), a  $H$  (méter) hosszal számolva:

$$T(F) = \frac{H}{V(F)}$$

A sebesség természetesen nem lehet (már csak számítástechnikai okokból sem) negatív, vagy nulla. Az ilyen értékek helyett automatikusan az 1 m/s lép be.

Modellünkben a forgalmi áram egységjárművekből áll, az áramlás sebessége egyenletes, minden egységjármű ugyanazzal a sebességgel halad. Ez az útkereséshez szükséges idő számításánál másképpen el sem képzelhető. Egyes sebességfüggő további adatok (pl. kilométerenkénti üzemeltetési költség) és eredmények rétegenkénti számításánál viszont egy további, rétegfüggő  $0.2 \leq \gamma \leq 1$  sebesség szorzófaktor is alkalmazunk. Ezt elméletileg úgy indokolhatjuk, hogy az egyenletes egységjármű-sebesség egy statisztikai átlaga a különböző rétegekhez tartozó járművek sebességének.

A csomóponton való áthaladás (csomóponti késleltetés) ideje függ a csomópont típusától, az élkapcsolat jellegétől (ld. 5.2.2.1.1.pont) az élkapcsolat forgalmától és a csomópont egyéb forgalmi áramlásaitól. Az alkalmazott számítási eljárás hosszadalmas és bonyolult, és mivel csak tágabb értelemben tartozik a dolgozat témájához, csak egy viszonylag egyszerű, de nem triviális példát közlök. További részleteket az [54] tanulmány tartalmaz. Az itteni példa: *táblás* elsőbbség szabályozású csomópontban egy *főirányból való balrafordulás* esete.

Legyen az  $\alpha = 0.000807$  stop táblás és  $\alpha = 0.000647$  nem stop táblás csomópontban, valamint  $\beta = 5000 * \alpha$ . Ezek elméleti és gyakorlati *forgalomszimulációból* adódó tényezők. Vezessük be a következő jelöléseket:

$F_f$  az aktuális irányhoz képest fölérendelt irányok forgalmának összege (E/ó)  
 $F$  az aktuális irány a forgalma (E/ó)  
 $x, Y$   $x = F_f * \alpha$ ,  $y = 2 * \alpha * (F_f + F)$   
 $g$   $g = \frac{F}{e^x * (e^x - 1)}$

Az élváltás másodpercben mért idejét jelölje  $T(F)$ .

$T(F) = \beta$  ha  $x < 0.1$   
 $T(F) = \infty$  ha  $x > 10$   
 $T(F) = \infty$  ha  $0.1 \leq x \leq 10$  és  $g \leq F$   
 $T(F) = \frac{3600}{g - F} * (1 - e^{-Y})$  ha  $0.1 \leq x \leq 10$  és  $g > F$

• **Költségszámítás**

Egy út megtételének költsége az útvonalba eső hálózati elemek - élek és élkapcsolatok - költségének összege. A költségelemek *rétegfüggők* is, a hálózati és rétegjellemzők együttesen alakítják ki őket. A költségszámítás feltételezi az előzetes időszámítást, lévén hogy egy jelentős költségelem az *idő értéke*. Maga az aktuális forgalom nagyság is csak az időn keresztül jelenik meg a költségben. A költséget egy általános  $G$  egységben mérjük.

Az egy élre és egy, az adott réteghez tartozó utazásra vonatkoztatott  $K(F)$  élköltséget, valamint ennek egy egységjárműre vetített  $KE(F)$  hányadát az alábbi levezetés szerint számítjuk:

H	élhossz (m)
T(F)	élidő (s)
ZK	idő egyenérték (G/ó), rétegfüggő időérték tag (G)
$KZK(F) = \frac{T(F)}{3600} * ZK$	
bkf	üzemköltségfaktor(%), útkategóriafüggő
bkv	üzemköltségfaktor, réteg és sebességfüggő
BK	üzemköltség (G/km), rétegfüggő üzemköltség tag (G)
$KBK(F) = \frac{bkf}{100} * \frac{H}{1000} * BK * bkv$	
gmf	útdíj faktor(%), útkategóriafüggő
gbf	útdíj kedvezmény faktor(%), útkategóriafüggő
GM	útdíj (G/km), rétegfüggő
GB	útdíj kedvezmény (G/km), rétegfüggő útdíj tag (G)
$KGM = \frac{H}{1000} * \frac{gmf}{100} * GM$	
$KGB = \frac{H}{1000} * \frac{gbf}{100} * GB$	útdíj kedvezmény tag (G)
tmf	kapudíj faktor(%), útkategóriafüggő
tbf	kapudíj kedvezmény faktor(%), útkategóriafüggő

TM	kapudíj (G/km), él és rétegfüggő
TB	kapudíj kedvezmény (G/km), él és rétegfüggő
$KTM = \frac{tmf}{100} * TM$	kapudíj tag (G)
$KTB = \frac{tbf}{100} * TB$	kapudíj kedvezmény tag (G)
pe	egységjáromű tényező, rétegfüggő
$K(F) = KZK(F) + KBK(F) + KGM - KGB + KTM - KTB$	
$KE(F) = \frac{1}{pe} (KZK(F) + KBK(F) + KGM - KGB + KTM - KTB)$	

Az élköltség természetesen nem lehet negatív, ehelyett automatikusan nulla lép be.

A csomóponti költséget egy áthaladásra egy, az adott réteghez tartozó utazásra vonatkoztatva egyszerűen az idő értékéből és az esetleges kapudíjból számítjuk.

- *Utak közti megosztás*

Ha két minimális utat veszünk figyelembe, akkor az  $r_1$  ellenállású első út és az  $r_2$  ellenállású második út között a ráterhelendő  $f$  forgalmat a következő függvény szerint osztjuk meg  $f_1$  és  $f_2$  részekre:

Vezessük be az  $r = \left( \frac{k * (r_2 - r_1)}{r_1} \right)^2$  jelölést. Ezzel:

$$f_1 = f \quad \text{és} \quad f_2 = 0 \quad \begin{array}{l} \text{ha } r_2 > 3 * r_1 \\ \text{ha } r_2 \leq 3 * r_1 \end{array}$$

$$f_2 = f * \frac{e^{-r}}{1 + e^{-r}} \quad \text{és} \quad f_1 = f - f_2$$

A  $k$  egy ráterhelési lépcsőnként megadható megosztási paraméter, alapértéke  $k = 2.8$  a programban. A megosztási a 3.3.1. pontbani általános képlet egy speciális esete.

- *Körzet-pont mátrixtranszformáció*

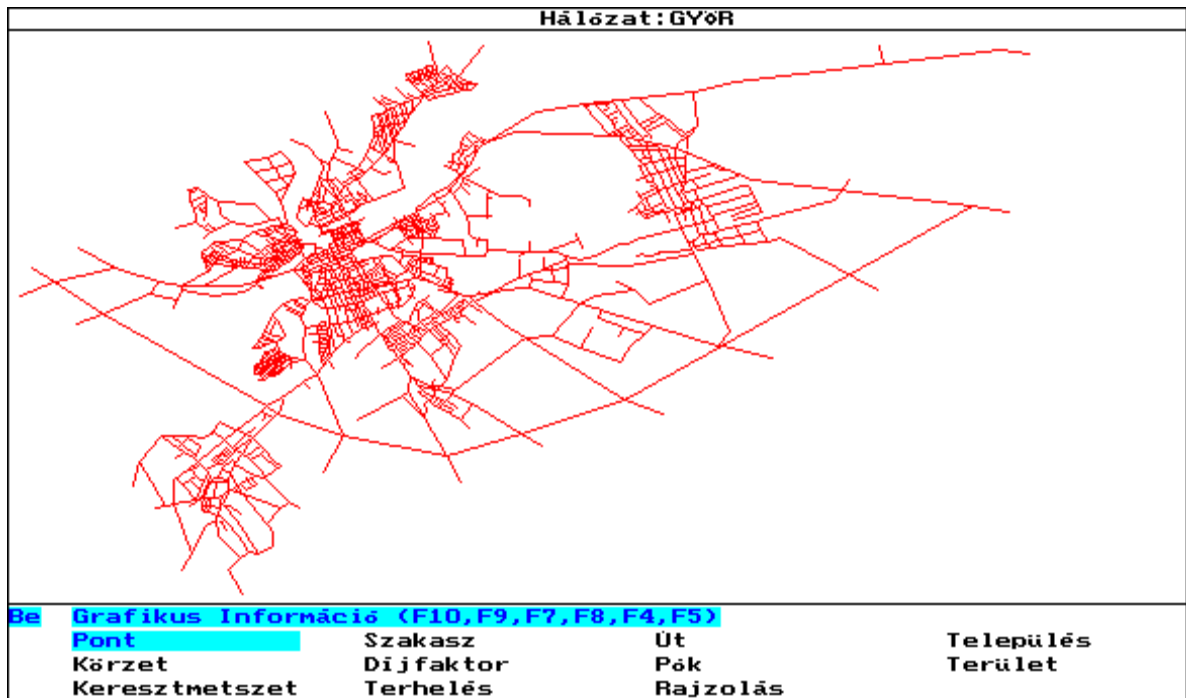
Egzaktan meghatározható útvonalak a hálózatban csak pontok között léteznek. Ezért szükséges a körzetmátrixoknak a forrás/nyelő pont és súly rendszernek megfelelő felbontása, a forrás-nyelő körzet forgalom forrás-pont-nyelőpont forgalmakká való átalakítás, vagyis a pontmátrixok létrehozása. Minden  $f/ny$  pont súlyával egyenesen arányosan részesedik a körzet forgalmából.

### 5.3.2. Információ és karbantartás

A rendszer a vele létrehozott és feldolgozott tervezési szituációkról teljeskörű információt nyújt mind az alapadatok, mind az eredmények vonatkozásában. Az információ az adat jellegűtől függően lehet szöveges vagy grafikus, vagy mind a kettő. A szöveges információ megjeleníthető a képernyőn (ld. pl. 27. és 28. ábra), de minden ilyen információ automatikusan megjelenik szövegfájlokban is, ahonnan esetleg továbbfeldolgozható, formázható, kinyomtatható.

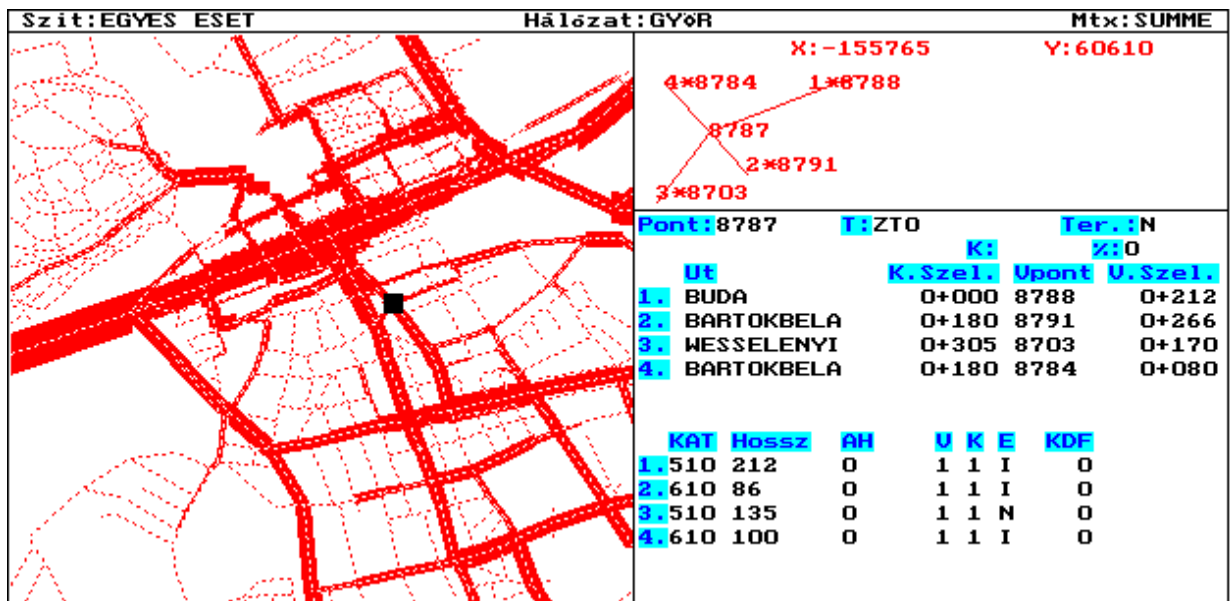
Grafikus információ kapható minden olyan adatról, amely közvetlenül (mint pl. egy pont jellemzői) vagy közvetve (mint pl. egy körzetben lévő forrás/nyelő pontok súlyelosztása)

köthető a hálózathoz, a hálózat grafikusan megjeleníthető elemeihez. A 33. ábra egy teljes hálózati "távlati" képet mutat.



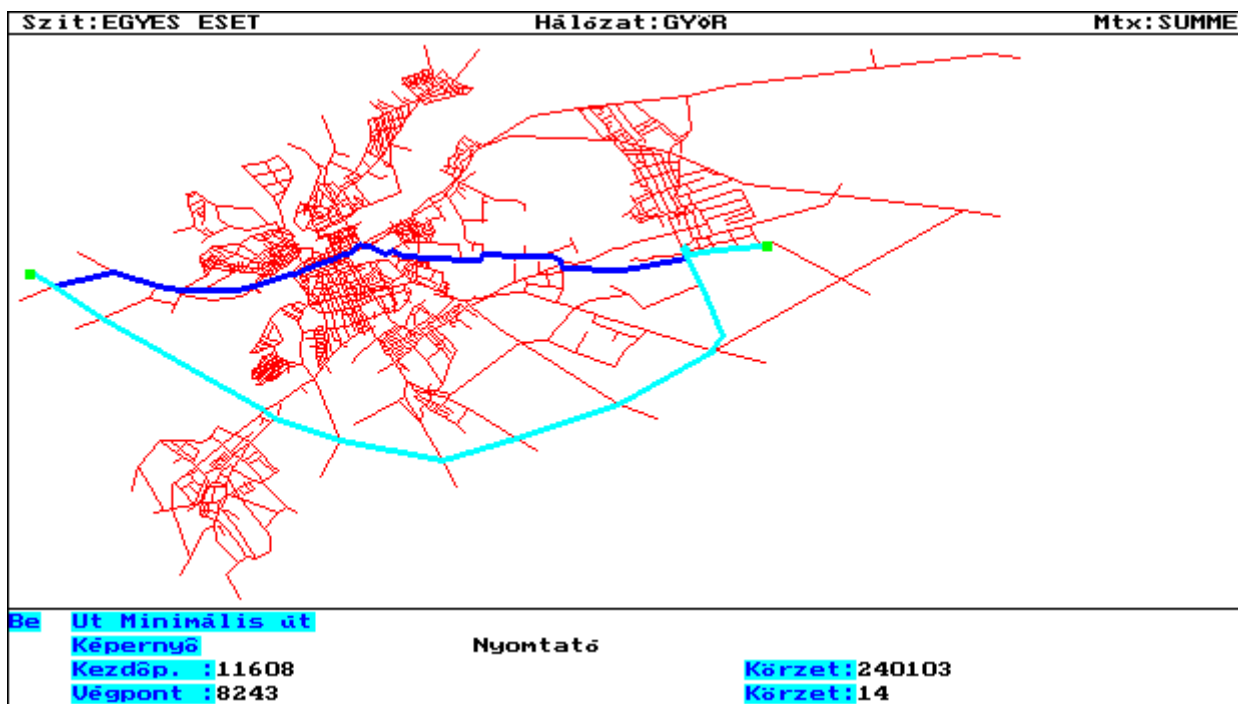
33. ábra. Teljes hálózat

A 34. ábrán ennek egy kinagyított része van, élterhelésekkel és a kép másik oldalán egy a nagy képen kijelölt pont "lokális" képével és adatai egy részével. A 35. ábrán egy elsősorban demonstrációs célokra használható információs funkciót láthatunk: ugyanazon két pont közti, kétféle ellenállásfüggvénnyel (időben és távolságban) számított minimális utakat. A kép "mögötti", egy billentyűváltással elérhető szöveges információ egy részét a 36. ábra mutatja. A minimális utak keresését terhelt hálózaton végeztük.



34. ábra. Terhelések





35. ábra. Minimális utak/1

Hálózat: GYŐR      Kezdőpont: 11608      Végpont: 8243

Minimális út távolságban

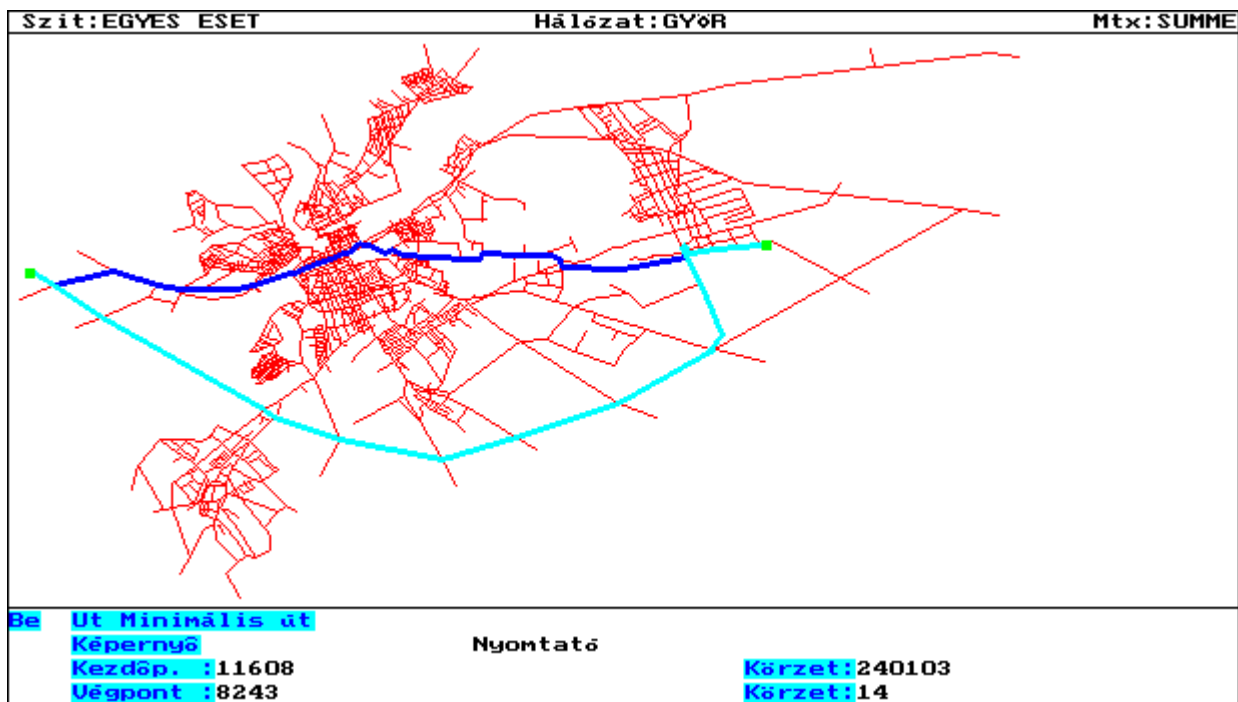
Út	Kezdő Szelv.	Vég Szelv.	Hossz km	Idő ó. p.mp	Seb. km/ó
VASUTSOR	1+540	1+065	0.475	0 59	29.0
VASUTSOR	1+065	0+598	0.467	0 58	29.0
VASUTSOR	0+598	0+271	0.327	0 41	28.7
10	131+452	132+492	1.040	0 58	64.6
85	0+000	1+392	1.392	1 17	65.1
M1	0+645	0+000	0.645	0 22	105.5
			17.776	25 49	

Minimális út időben

Út	Kezdő Szelv.	Vég Szelv.	Hossz km	Idő ó. p.mp	Seb. km/ó
VASUTSOR	1+540	1+065	0.475	0 59	29.0
VASUTSOR	1+065	0+598	0.467	0 58	29.0
VASUTSOR	0+598	0+271	0.327	0 41	28.7
M1	6+464	2+064	4.400	2 37	100.9
M1	2+064	0+645	1.419	0 51	100.2
M1	0+645	0+000	0.645	0 22	105.5
			22.833	21 02	

36. ábra. Minimális utak/2

A 37. ábrán egy kijelölt útvonal terhelését emeljük ki. A képhez tartozó és egy váltással elérhető teljes szöveges információ egy része a 38. ábrán látható.



37. ábra. Egy útvonal terhelése/1

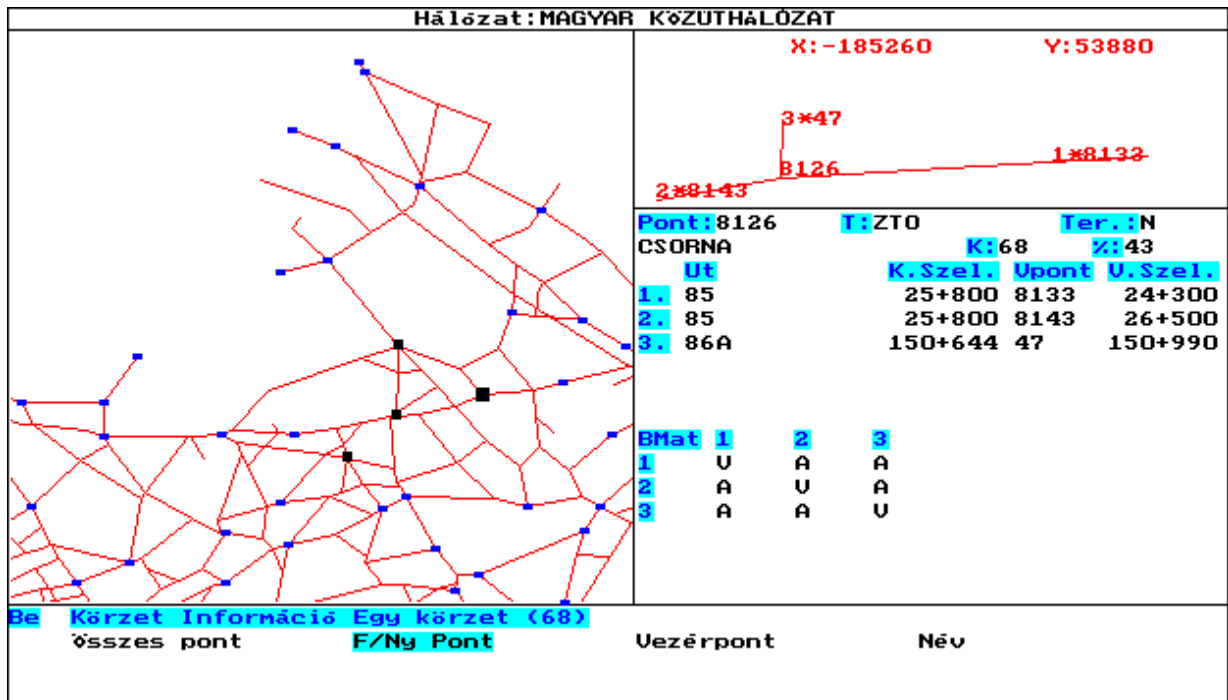
Hálózat :GYÖR Út:83 Terhelési lista  
 Szituáció:EGYES ESET Összeg (E/ó)

Kezdő Szelv.	Vég Szelv.	Hossz km	Idő ó. p.mp	Seb. km/ó	Út kat.	Szakasz terh. (E/ó)
0+000	0+100	0.100	0 5	72.0	170	71.539
0+100	1+559	1.459	1 17	68.2	170	52.194
1+559	2+215	0.656	0 35	67.5	170	75.874
2+215	2+958	0.743	0 39	68.6	170	75.874
2+958	3+803	0.845	0 44	69.1	170	115.111
3+803	4+203	0.400	0 21	68.6	170	115.894
.....						
5+720	6+210	0.490	0 26	67.8	120	208.501
6+210	6+610	0.400	0 21	68.6	120	212.683
6+610	7+035	0.425	0 22	69.5	120	197.789
7+035	7+360	0.325	0 25	46.8	520	201.862
7+360	7+759	0.399	0 31	46.3	520	127.548
7+759	8+210	0.451	0 35	46.4	520	136.510
8+210	8+342	0.132	0 10	47.5	620	129.866
8+342	8+518	0.176	0 14	45.3	620	129.866
-----						
8.518			08 05			

Átlagos szakaszterhelés (E/ó): 120.677  
 Összes forg. telj. (EKm/ó) :1027.923

38. ábra. Egy útvonal terhelése/2

A 39. ábrán egy hálózat egy részletén ki van jelölve az összes forrás/nyelő pont, köztük (más színnel ill. árnyalattal) egy adott körzet f/ny pontjai, a jobb oldalon a körzet egyik (külön megjelölt) f/ny pontjának adatai.



39. ábra. Egy körzet

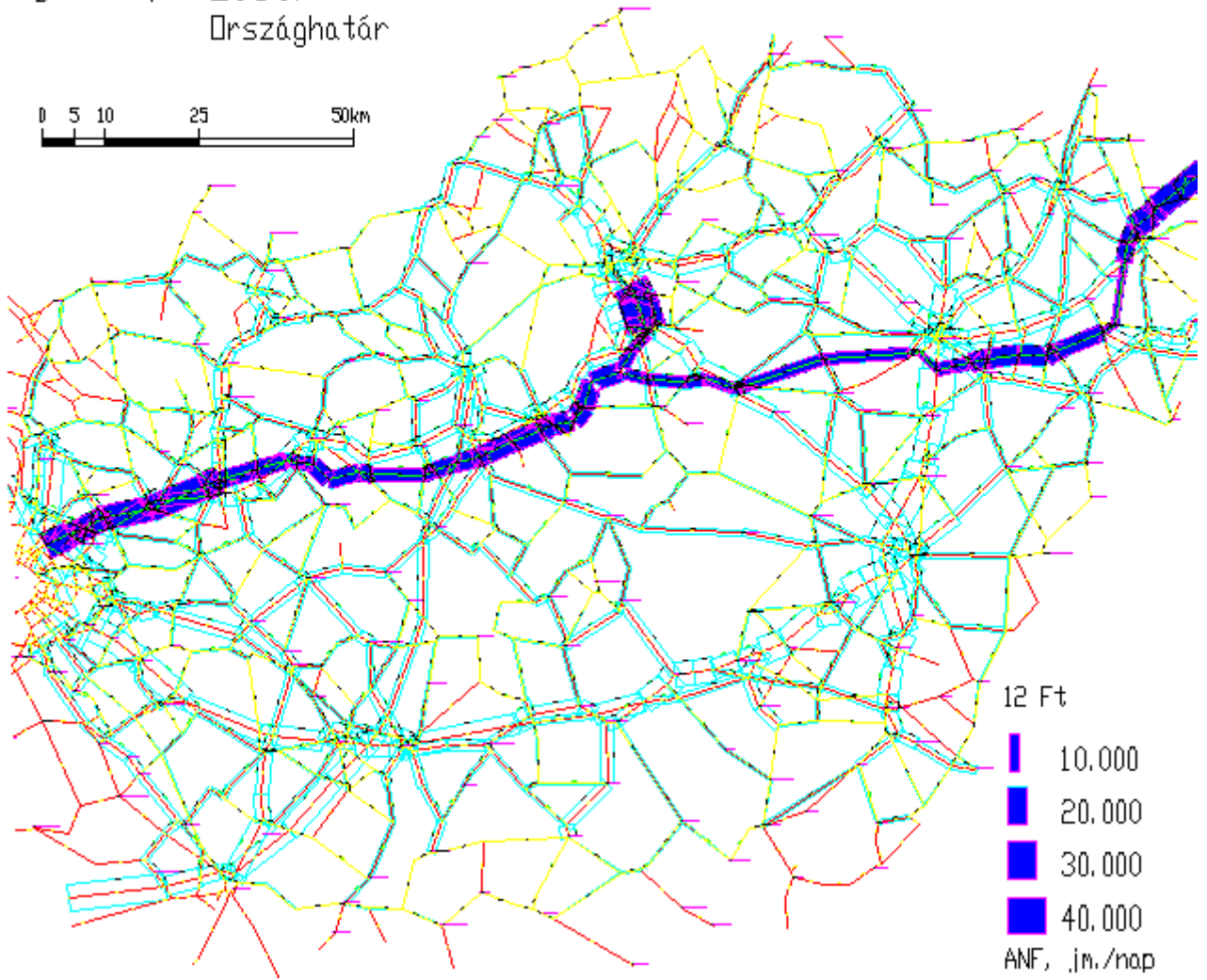
Az információszolgáltatás fontos komponense az, hogy az összes grafikus információt képes a rendszer DXF formátumú fájlalba kivinni, ahonnan megfelelő grafikus rendszerrel továbbfeldolgozhatók, kirajzolhatók, kinyomtathatók. Ilyen, AUTOCAD rendszerben való továbbfeldolgozással kapott grafikus eredményeket mutatunk be a 40-42. ábrákon.

A rendszer alapadatainak teljeskörű *karbantartását* elvégezhetjük a rendszeren belül. A *szöveges* jellegű karbantartás részben interaktív módon, részben egy, minden adatfajtura kiterjedő beépített *speciális szövegszerkesztővel* történhet. Ez az adatrögzítésnél elvégzi a lehetséges formai és tartalmi ellenőrzéseket is. *Külső* adatforrásból szövegfájlok formájában vehetünk át információt. Ez tipikusan a forgalmi mátrixoknak a rendszerbe való bekerülési módja, ezeknek a kiinduló változata általában valamilyen forgalomszámlálási, felmérési, statisztikai adatgyűjtés eredménye.

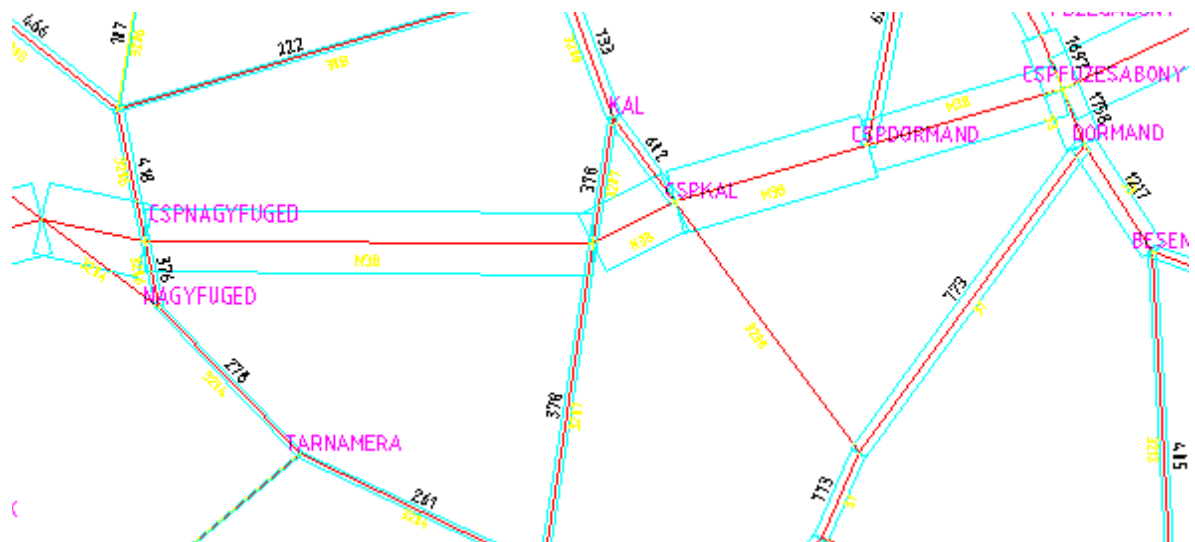
A *grafikus* felületen végezhető *interaktív* ("látható") karbantartás jogos felhasználói igény minden olyan adatról, amely a hálózathoz kötődik. Ezt rendszerünk teljeskörűen biztosítja, minden hálózati elem, és az ezekhez tartozó legtöbb adat felvihető, törölhető, módosítható a grafikus programkomponenssel. A grafikus lekérdezés és karbantartás hatékonyságát, biztosítja a beépített automatikus *naplózás*, amely tárolja, megjegyzi az elvégzett módosításokat, lehetővé téve ezzel az egyes változatok sorában való előre és vissza lépkedést, a "visszavonás" funkciót, korábbi változatok egyszerű, ismételt munkavégzés nélküli visszaállítását.



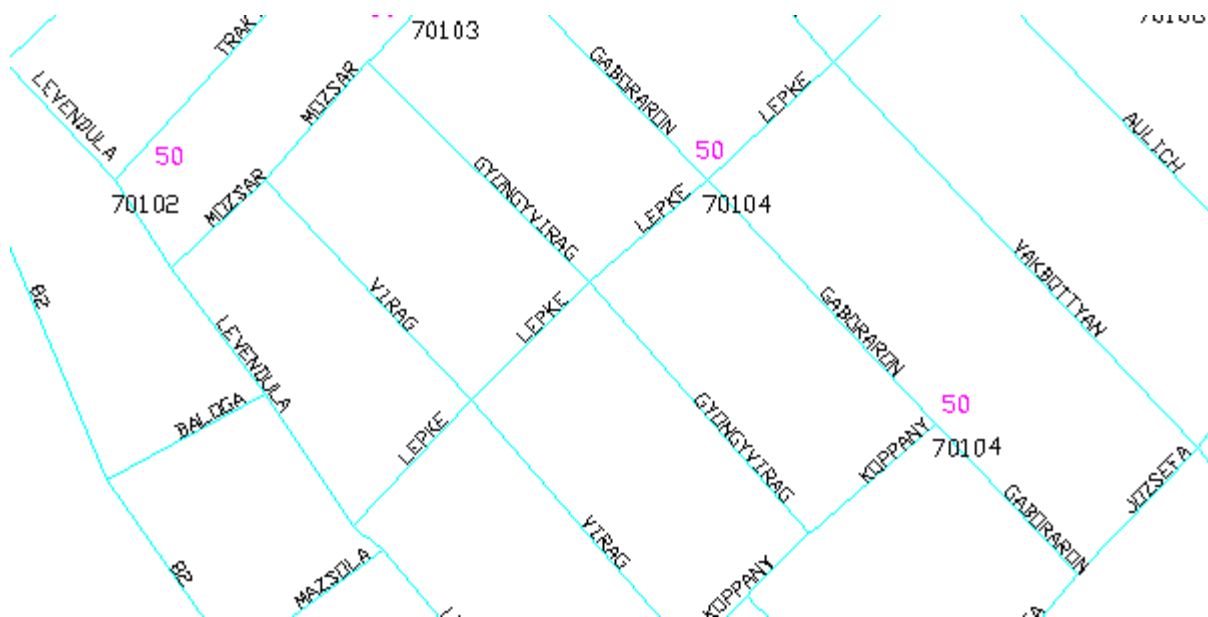
M3 autópálya  
2030.  
Országhatár



40. ábra. M3 teljes



41. ábra. M3 részlet



42. ábra. Győr részlet

### 5.3.3. Segédfunkciók

Az itt ismertetendő funkciók a rendszer fő közlekedéstervezési szolgáltatásaihoz használatos fontos segédeszközök.

- *Előtöltés*

Minden olyan esetben, amikor útvonalat keresünk és választunk, az eredmény függ a hálózati elemek aktuális ellenállásától, vagyis a hálózat aktuális terhelésétől. A ráterhelés mellett a *részterület kiemelés* és a *kalibráció* is tartalmaz útvonalkeresést. Természetesen jobban modellezzük a valóságot, ha nem az "üres" (terhelés nélküli) hálózaton, hanem valamely terhelt hálózaton végezzük el az útvonalválasztást. Erre a célra szolgál az *előtöltés* segédfunkció. Rendszerünkben kétféle előtöltést értelmezünk:

- *Szituációs előtöltés*: A hálózati elemeket valamely, már rendelkezésünkre álló ráterhelési eredménnyel (választhatóan a teljes forgalommal vagy ennek bizonyos összetevőivel, rétegeivel) töltjük fel.
- *Kategóriás előtöltés*: A hálózati elemeket az útszakaszok *útkategória* jellemzője által meghatározott *elvi forgalmi kapacitás* adat bizonyos (kategóriánként vagy globálisan megadható) százalékának megfelelő forgalommal töltjük fel.

A kétféle előtöltés együtt is alkalmazható.

- *Szimmetrizáció*

A modellezendő valóságban több lényeges elem eleve - a közlekedéstervezői megfontolásoktól függetlenül - *szimmetrikus*. (Gondoljunk pl. arra, hogy a kétirányú útszakaszok esetén teljesen általános a két irányban azonos számú forgalmi sáv létesítése, az esetlegesen aszimmetrikus igénybevétel ellenére is.) Bizonyos esetekben hasznos lehet a kiinduló vagy eredményadatok szimmetrizálása. Rendszerünkben *szimmetrizálni* lehet:

- A forgalmi *mátrixokat*. Az eredmény a főátlóra szimmetrikus, a szimmetrikusan elhelyezkedő elemek mindegyike az eredetiek számtani átlaga lesz.
- A hálózati elemek forgalmi *terhelésadatait*. A kétirányú szakaszok terhelése irányonként azonos, az eredeti irányonkénti terhelések számtani közepe lesz. A

csomóponti terhelési mátrixokat az így létrejött forgalmakhoz igazítjuk (az egyirányú szakaszok miatt ezek nem feltétlenül lesznek szimmetrikus mátrixok).

- *Mátrixkiegyenlítés*

A kiindulási adatként szereplő *forgalmi mátrixokat* sok esetben csak úgy tekinthetjük, mint egy (statisztikai értelemben vett) mintát, amit bizonyos előírt peremfeltételekhez hozzá kell igazítani, amit egy előzetes kalibrálás, *kiegyenlítés* tesz megfelelőbbé, használhatóbbá a modellezésben. A mátrixkiegyenlítés általános, hálózat és forgalomfüggő megoldása a rendszer egyik fő funkciója, később (5.3.5.) tárgyalom. Ennek egy speciális esete a *sor és oszlopszorzásos kiegyenlítés* önállóan, segédfunkcióként is használatos rendszerünkben. Az algoritmus egy iteratív eljárás, amely a mátrixot ismételt sor és oszlop szorzásokkal (adott hibahatáron belül) olyanná alakítja, amelynek sor és oszlopösszegei (adott hibahatáron belül) megegyeznek előre rögzített értékekkel (peremfeltételek). Az eljárás matematikai elemzését, a megoldhatóság (konvergencia) feltételeit illetően a szakirodalomra [23], [35] utalok. Az eljárás egyben egy körzetenkénti *két növekedési tényező*s forgalomelőrebecslési szerepkörben is használatos a rendszerben.

- *Összevetés*

Bizonyos esetekben sok információ kapható, sok következtetés levonható, ha nem csak önmagukban az egyes tervezési situációkat vizsgáljuk, hanem különböző situációk *viszonyát* is. Rendszerünkben lehetséges a terhelési eredmények ilyen jellegű vizsgálata, olyan módon, hogy meglévő situációkból *különbségsituációkat* képezünk a forgalmi terhelések különbségével. Az összevetendő két situáció akármilyen lehet, a hálózatok azonossága, vagy tartalmazási viszonya sem megkövetelt. Az eredmény a két hálózat metszetén keletkezik és minden olyan eredményadatot származtatunk, amely a különbségsituációra is értelmezhető.

### 5.3.4. Részterület kiemelés

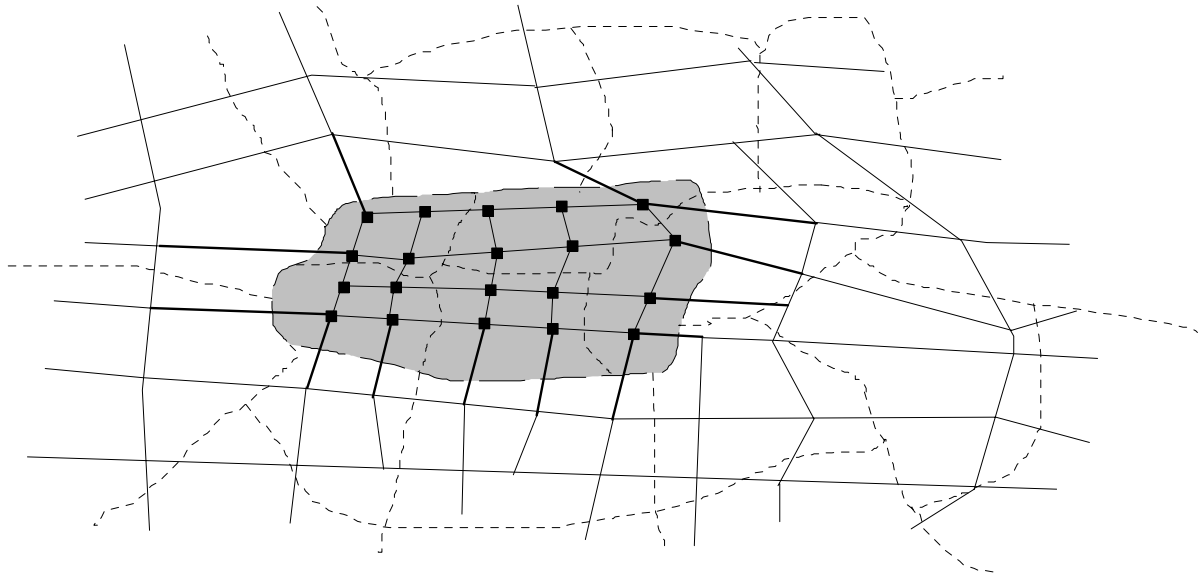
A modellezendő közlekedési hálózatok általában nem izoláltak. A bennük lezajló forgalom jelentős része lehet *külső eredetű*, a hálózaton kívül keletkező és/vagy elnyelődő, de a hálózat forgalmi viszonyait jelentősen befolyásoló forgalmi összetevő. Más megfogalmazásban ezt úgy mondhatjuk, hogy egy adott, valamilyen szempont szerint lehatárolt közlekedési hálózat mindig egy őt tartalmazó nagyobb hálózat *részhálózata*, a lehatárolás mindig egy a nagy hálózatból való *kiemelést* jelent.

Az ilyen kiemelés a tervezési gyakorlat természetes eszköze, hiszen mindig a "teljes" hálózattal dolgozni mind az emberi munkát, mind a számítógép-kapacitást tekintve olyan jelentős többletráfordításokat igényel, amelyek egy jól megtervezett *részterület kiemeléssel* megtakaríthatók, az eredmények romlása nélkül.

Ezt a tervezési lépést támogatja rendszerünk egyik fő funkciója, a *részterület kiemelés*. A funkció leírásához nevezzük az eredeti, kiinduló hálózatot *nagy* hálózatnak, a kiemelt, eredmény hálózatot *kis* hálózatnak.

A funkció *kiinduló adatai*:

- A *nagy hálózat* adatai, a *körzetbeosztással* és egy erre vonatkozó *forgalmi mátrix*.
- A *kis hálózatba* bevonandó *körzetek*, nevezzük őket *belső* körzeteknek. A kiemelés határa egyben mindig körzethatár is.
- A *kis hálózat* pontjai, nevezzük őket *belső* pontoknak és a *kordonszakaszok*, amelyek mindig *belső* és *külső* (nem *belső*) pontokat kötnek össze.



43. ábra: Részterület kiemelés

Illusztrációképpen tekintsük a 43. ábrát. Az ábrán a szaggatott vonal kis hálózat határa, egyben körzethatár is, a pontozott vonalak körzethatárok, a megjelölt pontok a belső pontok, a vastagított szakaszok a *kordonszakaszok*.

A funkciónak két fő *eredménye* van, ezek:

- A *kis hálózat* mint önálló hálózat, a saját *körzetbeosztásával* és egy erre vonatkozó *forgalmi mátrix*. A kis hálózat körzetei egyrészt az eredeti belső körzetek lesznek, másrészt új körzetekként belépnek az egyetlen *forrás/nyelő* pontból álló - a kordonszakaszokból keletkező - *kordonkörzetek*, amelyek a nagy hálózatban a megfelelő kordonszakaszon belépő/kilépő forgalmat bocsátják ki, illetve nyelik el.
- A kis hálózathoz tartozó, ennek új körzetbeosztásához illeszkedő *forgalmi mátrix*, amelyet a kis hálózatra ráterhelve ugyanolyan forgalmi viszonyokat kapunk, mintha a nagy hálózatra az eredeti forgalmi mátrixot terheltük volna rá.

A funkció megoldó algoritmusának magja az *útvonalválasztás* (opcionálisan előtöltéssel). A nagy hálózat forrás/nyelő pontjai közti minimális utakat meghatározva és elemezve szétválogatjuk a nagy hálózat forgalmát a kis hálózat szempontjából több összetevőre:

- *Belső forgalom*: belső pontból indul, nem érint kordonszakaszt, és célpontja is belső pont.
- *Kiinduló forgalom*: belső pontból indul, egy kordonszakaszon lép ki a kis hálózatból, célpontja külső pont.
- *Cél forgalom*: külső pontból indul, egy kordonszakaszon lép be a kis hálózatba, célpontja belső pont.
- *Átmenő forgalom*: külső pontból indul, egy kordonszakaszon belép a kis hálózatba, egy másik kordonszakaszon elhagyja ezt, célpontja külső pont.
- *Elkerülő forgalom*: külső pontból indul, célpontja külső pont és nem érint belső pontot.

Ezekből az összetevőkből képezzük az új mátrixot. Természetesen létezhet a fentiekbe be nem sorolható "*szabálytalan*" útvonal is, például ha belső pontból indul és ugyancsak belső pontban végződik, de két vagy több kordonszakaszt is érint. A rendszer ezt figyelni és

szabálytalanság esetén *figyelmezteti* a tervezőt a lehatárolás pontatlanságára, lokalizálva az ajánlott korrekció helyét is.

Megjegyzem még, hogy a funkció formailag hasonló a korábban ismertetett *területi relativ forgalom* előállításához, számítástechnikailag sok közös elemet is tartalmaz ezzel, de vegyük észre, hogy itt a *tervezési cél* az, ami teljesen más.

### 5.3.5. Kalibráció

A rendszerben *kalibrációnak* nevezzük azt a tervezési lépést, amellyel körzetszintű forgalmi mátrixokat *kalibrálunk*, kiegyenlítünk, hozzá igazítunk egy hálózaton adott előírt élterhelési (pl. keresztmetszeti számlálási) adatokhoz, mint *peremfeltételekhez*. A kalibráció (más néven mátrixkiegyenlítés) elmélete és gyakorlata önmagában kimerítené egy dolgozat kereteit, itt csak nagy vonalakban a teljes rendszerismertetés kedvéért foglalkozom e témával. Bővebb és pontosabb tárgyalást illetően a szakirodalomra [38], [31], [39] utalok.

A funkció *kiinduló adatai*:

- A *hálózati* adatok a *körzetbeosztással* és egy erre vonatkozó *forgalmi mátrix*.
- Egy hálózati kiegészítő adat, a *keresztmetszet* adatállomány, amely egy élhalmaz, élenként megadott előírt terhelési adatokkal (peremfeltételek). Egy keresztmetszeten a továbbiakban is mindig egy élt értünk.

Illusztrációképpen nézzük a 44. ábrát. Az ábrán a kiemelt pontok forrás/nyelő pontok (**a...e**), a kiemelt (vastagított) szakaszok a keresztmetszet állomány elemei, minden szakasznál mind a két irányú él. A mátrix pontszintű forgalmi mátrix.

A funkció *eredménye*:

- A körzetszintű *forgalmi mátrix* kiegyenlített formája, amelyet a hálózatra visszaterhelve a kijelölt éleken az előírt terhelési adatokat kapjuk vissza.
- Mellékeredményként adódik egy pontszintű *forgalmi mátrix*, amelyet elemezve hasznos következtetéseket vonhat le a tervező a körzetenkénti *f/ny súlyrendszernek* az adott mátrixhoz (pontosabban réteghez) való igazodását illetően, esetleg korrigálhatja ennek alapján a súlyrendszert.

Az eredményekkel kapcsolatban megjegyzem, hogy ezeket nem feltétlenül kapjuk meg minden esetben pontosan, sokszor meg kell elégednünk egy *közellítéssel*. Ennek oka az, hogy a kalibrációs eljárás egy iterációs eljárás, amelynek konvergenciája a kiinduló adatoktól függ és elméletileg nem minden esetben garantált.

A funkció megoldó algoritmus egy iteráció, amely egy megadott pontosság, vagy adott lépésszám eléréséig folyik (45. ábra).

Az algoritmus fő elemei:

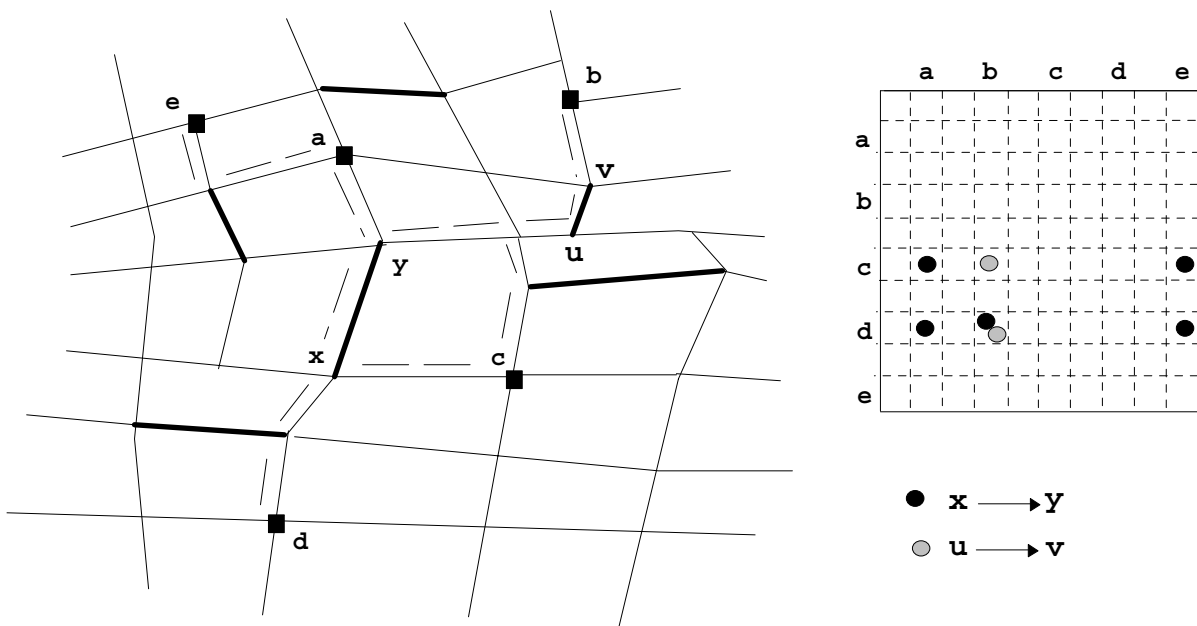
- *Összeválogatás*  
Az adott körzetszintű mátrixot pont szintűre bontjuk. Az (opcionálisan előtöltött) hálózaton minden *keresztmetszethez* meghatározzuk a *hozzárendelt* viszonylatokat, vagyis azokat a viszonylatokat, amelyek forgalma - az alkalmazott *útvonalválasztási* modell szerint - áthalad a keresztmetszeten. A 44. ábrán az (**x, y**) élhez és az (**u, v**) élhez rendelt viszonylatokat jelöltem be a mátrixban, feltéve, hogy egy minimális úttal



dolgozunk, és a feltüntetett viszonylatokban a minimális utak az ábrán szaggatott vonallal jelöltek.

- *Felszorzás*

Az egy keresztmetszethez hozzárendelt viszonylatok forgalma, tehát a pontmátrix megfelelő elemeinek összege *egyenlő* kell hogy legyen a keresztmetszetre *előírt* forgalommal. Ezt az egyezést a megfelelő mátrixelemek azonos értékkel való megszorzásával állítjuk be. Nevezzük ezt az értéket *keresztmetszeti szorzónak*. A szorzásokat keresztmetszetenként, ezek valamilyen sorrendjében hajtjuk végre. Mivel az egyes keresztmetszetekhez rendelt viszonylatok halmazai *átfedést* is tartalmazhatnak (ld. az ábrán a **d-b** viszonylat), egyik szorzás elronthatja a másik által beállított egyenlőséget.



44. ábra. Kalibráció

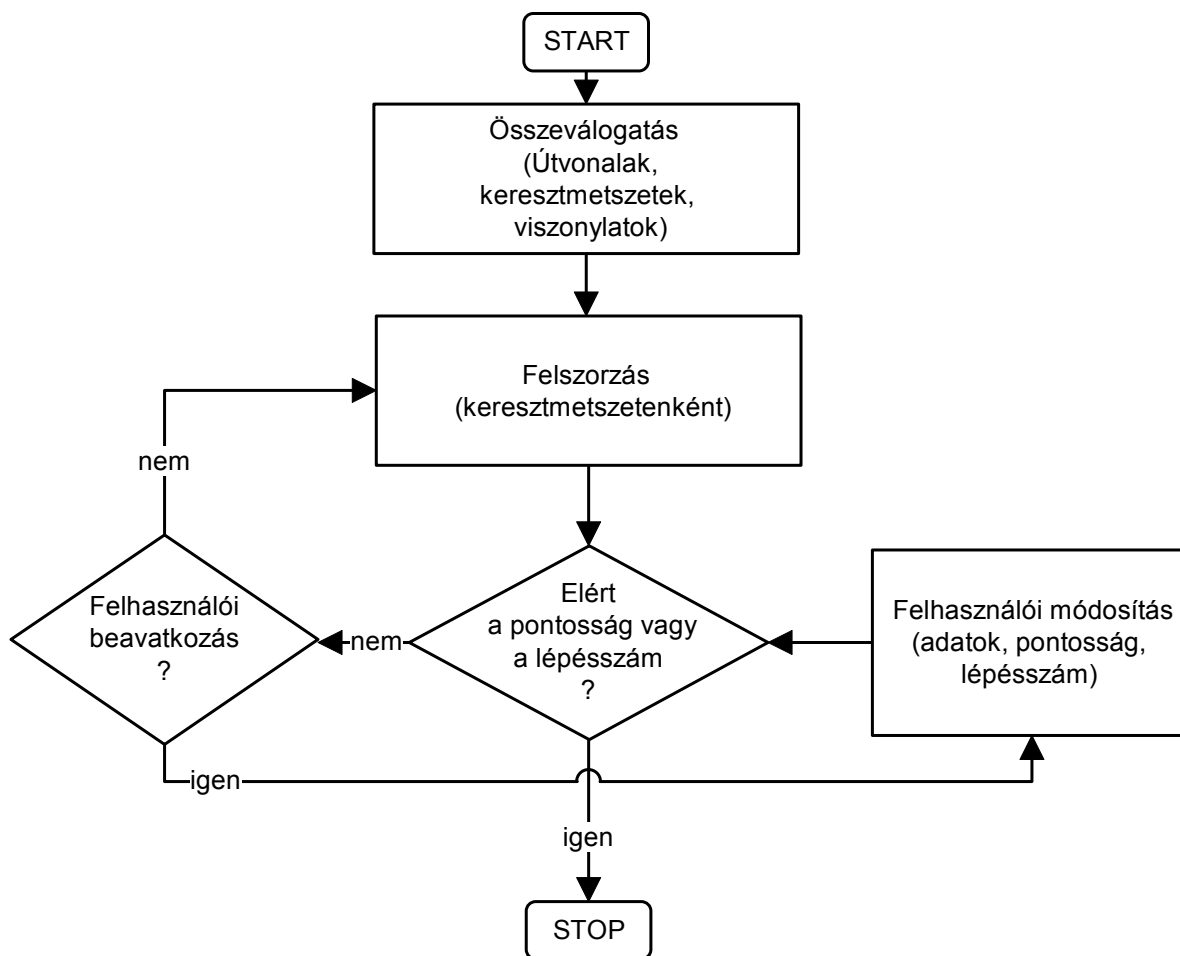
- *Ellenőrzés*

Aktuális pontosságon értjük a *keresztmetszeti szorzóknak* az 1 -től (abszolút értékben) való eltérési maximumát. Ha ez nem nagyobb, mint az előírt pontosság, vagy a *felszorzás* lépés már az előírt lépésszámra végrehajtásra került, akkor vége az eljárásnak, összevonással kialakítjuk a pontmátrixból a körzetmátrixot és készen vagyunk.

- *Felhasználói beavatkozás*

Az eljárás nem mindig konvergens, tehát célszerű a menetközbeni felhasználói beavatkozás lehetősége. Ez a rendszerben több módon is szabályozható (adott billentyűre, adott számú felszorzási lépésenként stb.). A tervező itt - akár keresztmetszetenként is - tanulmányozhatja az elért helyzetet, törölhet, felvehet keresztmetszetet, módosíthatja az előírt forgalmakat, új pontosságot, új lépésszámot állíthat be.

Megjegyzem, hogy a rendszerben a *két utas kiegyenlítés* is lehetséges. Ekkor a forgalmakat megosztva két pontmátrix tartalmazza és a viszonylatok keresztmetszetekhez rendelésénél az első két minimális utat vesszük figyelembe.



45. ábra: Kalibrációs algoritmus

Számítástechnikai szempontból érdemes még megjegyezni, hogy a kalibráció rendszerünk leginkább *erőforrásigényes* modulja. A felszorzási ciklusban a mátrixot (vagy mátrixokat) mindenképpen az operatív tárban kell tartani mert különben a végrehajtási idő elviselhetetlenül nagy lesz, még gyors gépen is. A másik kritikus tényező a keresztmetszet-viszonylat hozzárendelések tárolási módja, hiszen erre is szükségünk van a felszorzási ciklusban. Szerencsés esetben (nagy operatív tárral rendelkező gép és nem túl sok hozzárendelés) ez is megoldható egy operatív tárbeli adatstruktúrával. Ha ezt háttértáron tudjuk csak kezelni, általában egy nagyságrenddel nagyobb futási időt kapunk.

### 5.3.6. Ráterhelés

A rendszer *ráterhelési* modulja a dolgozatban tárgyalt lépcsős ráterhelés egy megvalósítása. A folyamatban részt vesz (illetve opcionálisan részt vehet ) a rendszer teljes adatbázisa (ld. 5.2.pont). A ráterhelés algoritmusait korábban már részletesen tárgyaltam, itt inkább egy szituáció kiinduló és eredmény adatkörnyezetének pontosabb specifikálására törekszem.

Egy ráterhelési szituáció *kiinduló adatai*:

- A *hálózat* és a *körzetbeosztás* alapadatai.

- (opcionális) Előtöltési adatok:
  - A *situációs előtöltés* situációja, ezen belül az előtöltésben résztvevő rétegek vagy az összeg kijelölése.
  - A *kategóriás előtöltésben* résztvevő *útkategóriák* kijelölése és a feltöltési százalékok megadása.
- Az útvonalválasztás *optimalizálási szempontja* (a hálózati elemek ellenállásának számítása) lehet: *idő* vagy *költség* szerinti.
- Az útvonalválasztásnál figyelembeveendő *minimális* utak száma: lehet: *egy* vagy *kettő*.
- Az útvonalválasztásnál (az útvonal ellenállásának számításánál) figyelembe vesszük e a *csomóponti ellenállásokat* lehet: *igen* vagy *nem*.
- (opcionális) *Területre* vonatkozó *relatív* forgalom meghatározás (4.3.1.pont) adatkijelölése.
- (opcionális) *Élekre* vonatkozó *relatív* forgalom meghatározás (4.3.2.pont) adatkijelölése.
- (opcionális) *Részhálózati* forgalom meghatározás (4.5.pont) adatkijelölése
- (opcionális) *Útdíjas autópályák* díjszámításával (4.4.pont) kapcsolatos adatkijelölés.
- A ráterhelési situációban résztvevő *forgalmi rétegek* globális rétegjellemzői.
- *Ráterhelési lépcsők száma*. *Lépcsőnként* megadandó:
  - A *csoporthármatrix* kijelölése.
  - A csoportmátrix *rétegzonosítója* (melyik réteghez tartozik).
  - A százalékos *arány* (a csoportmátrix forgalmának hány százalékát vesszük itt figyelembe).
  - A *súlyrendszer* (az f/ny pontok melyik súlyrendszerét vesszük itt).
  - (opcionális) Forgalmi tiltások (ld. 4.2. .pont) adatainak kijelölése.
  - *Megosztó* paraméter (két út figyelembevétele esetén)

Egy ráterhelési situáció *eredmény adatai*:

- *Élenként*
  - forgalmi terhelés
  - terhelt élidő
  - terhelt sebesség
  - sebességvesztés
  - forgalmi teljesítmény
  - kapacitáskihasználtság
  - összes élidő
  - összes és átlagos idővesztés
  - összes üzemeltetési költség
  - összes időköltség (idő egyenérték)
  - összes kilométerdíj (autópálya)
  - összes kilométerdíj kedvezmény (autópálya)
  - összes kapudíj (autópálya)
  - összes kapudíj kedvezmény (autópálya)
  - összes élköltség (költségösszeg)
- *Szakaszonként*
  - a szakasz éleinek adatai értelemszerűen összeadva ill. átlagolva.
- *Csomópontonként*
  - csomóponti forgalmi terhelési mátrix
  - csomóponti terhelt idő mátrix
  - csomóponti költség mátrix
  - összes forgalmi terhelés

- összes idő
- összes költség
- összes és átlagos idővesztés
- *Útvonalanként*
  - az útvonalba eső élek és csomópontok adatai értelemszerűen összeadva ill. átlagolva. Egy útvonal kijelölésére többféle lehetőség van:
    - útzonosítóval (útszám, opcionálisan kezdő és végszelvény)
    - két pont között úthosszban minimális útként
    - két pont között időben minimális útként
    - két pont között költségben minimális útként
- *Útkategóriánként*
  - az adott útkategóriájú szakaszok adatai értelemszerűen összeadva ill. átlagolva.
- *A teljes hálózatra*
  - összes és átlagos életterhelés
  - összes és átlagos forgalmi teljesítmény
  - átlagos kapacitáskihasználtság
  - összes és átlagos élidő
  - átlagos sebesség
  - átlagos sebességvesztés
  - összes és átlagos élidővesztés
  - összes és átlagos csomóponti idő
  - összes és átlagos csomóponti idővesztés
  - összes és átlagos idővesztés
  - összes utazási idő
  - összes csomóponti költség
  - összes időköltség
  - összes üzemeltetési költség
  - összes kilométerdíj (autópálya)
  - összes kilométerdíj kedvezmény (autópálya)
  - összes kapudíj (autópálya)
  - összes kapudíj kedvezmény (autópálya)
  - összes élköltség
  - teljes utazási költségösszeg
  - utazási távolságok eloszlása (grafikonban is)
  - utazási idők eloszlása (grafikonban is)
  - utazási költségek eloszlása (grafikonban is)
  - kapacitáskihasználtság eloszlása (grafikonban is)
- *A kijelölt részhálózatra*
  - ha van részhálózati (4.5.pont) kijelölés ennek viszonylataira is megkapjuk az összes terhelési, idő és költségadatokat.
- *A kijelölt területre*
  - ha van területi relatív forgalmi (4.3.1.pont) kijelölés, akkor az összes eredményadatot megkapjuk a *belső, kiinduló, cél, átmenő* és *elkerülő* bontásban is.
- *A kijelölt élekre, szakaszokra*
  - ha van ilyen relatív forgalmi (4.3.2.pont) kijelölés, akkor az összes eredményadatot megkapjuk csak erre vonatkozóan is.
- *A forgalmi rétegekre*

- az összes, rétegre is értelmezhető, adat megkapható forgalmi rétegenkénti bontásban is.

### 5.3.7. Technikai funkciók

Rendszerünk, mint minden nagyméretű, összetett és bonyolult programrendszer sok, tisztán a könnyebb, egyszerűbb és hatékonyabb felhasználást szolgáló technikai funkcióval is rendelkezik. A legfontosabbakat soroljuk fel az alábbiakban:

- *Rendszernapló*: a rendszer üzemeltetésének bizonyos, számítástechnikai szempontból fontos eseményeit jegyzi. Rögzítve van benne az egyes rendszer és programindítások ideje, a programokból való kilépési idők, az egyes programok befejezési módja, esetleges programhibák azonosító adatai.
- Tervezési szituációk létrehozása archív szövegfájlokból, kimentése szövegfájlokba.
- Tervezési szituációk, illetve ezek egyes részeinek másolása, törlése.
- Szövegfájlok karbantartása, beépített speciális szövegszerkesztővel.
- *Grafikus napló*: A grafikus tevékenységeket jegyzi, az egyes grafikus szituációk könnyebb, gyorsabb ismételhősége, visszaállíthatósága céljából.
- A grafikus adatok kiírása külső CAD programmal való továbbfeldolgozásra alkalmas formátumban.

## **6. Közlekedéstervezési programcsomagok**

### **6.1. Bevezetés**

Ebben a fejezetben áttekintjük az általam ismert, vagy legalábbis ismertelési szinten elérhetővé vált, a magyar *oktatási* vagy *tervezési* gyakorlatban a jelen időszakban (a kilencvenes években) előforduló olyan közlekedéstervezési programcsomagokat, amelyeknek lényeges részét képezi a *ráterhelési* feladat megoldása.

Előrebocsátom, hogy bár - a dolog erősen behatárolt szakmai jellegénél fogva - az ilyen programcsomagok száma nem nagy, a róluk - különösen az alkalmazásairól, az alkalmazási tapasztalatokról - való informálódás sokszor nehézségekbe ütközik. Ennek alapvető oka az, hogy ezek a programok és a hozzájuk kötődő alkalmazási-tervezési metodikák nemcsak kutatási-fejlesztési eredmények, hanem *termelőeszközök* is, amelyek beszerzése és hatékony használatuk megtanulása komoly anyagi és élőmunka ráfordításokat igényel. Mivel az alkalmazási lehetőségek száma nem nagy és a megrendelői kör erősen behatárolt - gyakorlatilag a *minisztériumi* és kisebb mértékben a nagyvárosi önkormányzati szint - az alkalmazás *versenyhelyzetben* folyik.

A fentiekkel magyarázható, hogy az alábbi áttekintés valószínűleg nem teljes és az ismertetés mélysége valamint részletessége is eltérő az egyes programcsomagokat tekintve.

### **6.2. TransCAD**

A *TransCAD Transportation GIS Software* programcsomag a *Caliper Corporation* (USA, Newton, Massachusetts) cég terméke. A programcsomag egy közlekedéstervezési szoftver, amely egy *térinformatikai* (GIS : Geographic Information System) szoftverkörnyezetbe van beágyazva [12], [13].

Az itt tárgyalt személyi számítógépes változat - amely 1996-os keltezésű és 3.0 verziószámú - Windows 3.1, Windows 95 és Windows NT operációs rendszer környezetben futtatható. A programcsomag egy 32 bites platformú, alapvetően C és C++ nyelven fejlesztett alkalmazás. A környezetből adódik, hogy a kezelhető feladatok nagyságára (pl. a hálózat méretére) vonatkozóan nem a szoftver, hanem a futtató gép memóriakapacitása és gyorsasága szab korlátot. A programcsomag használata hardver kulccsal védett.

Tekintve a GIS és Windows környezetet, a rendszer adatmanipulációs szempontból maximálisan felhasználóbarát, könnyen kezelhető. A közlekedéstervezési feladatok kiinduló adatbázisa a (természetesen megfelelően feltöltött) térinformatikai adatbázisból is előállítható, beleértve ebbe nem csak a pontokat éleket, távolságokat, hanem olyan gazdasági strukturális adatokat is, mint pl. egy körzet lakosainak száma vagy a személygépkocsik száma egy településen. A szinte mindenre kiterjedő grafikus adatmanipuláció a CAD rendszerekben megszokott szolgáltatási szinten realizálható (többretegűség, sokféle transzformáció stb.).

A rendszerről a Caliper Corporation cég World Wide Web (WWW) honlapjáról indulva is találhatunk ismertető, összefoglaló jellegű információkat.

A programcsomag pályázat eredményeként került a győri Széchenyi István Főiskolára 1997 - ben. Jelenleg előkészületek folynak arra, hogy - mint a tananyag része és tudományos diákköri munkák, diplomamunkák eszköze - a *szakirányú* valamint a *posztgraduális* és

egyetemi szintű képzésben való felhasználásra kerüljön. Más magyarországi felhasználóról jelenleg nincs tudomásunk.

A programcsomagon belül a ráterhelési modul is a teljességre törekszik, a jelenleg ismert főbb modellcsoportok mindegyike képviselve van. Az implementált módszerek:

- *Mindent vagy semmit eljárás*
- *Lépcsős ráterhelés*
- *Determinisztikus egyéni optimum*
- *Sztohasztikus egyéni optimum* (Sheffi [66] módszer)
- *Rendszeroptimum*

A élköltség alapesetben időben van kifejezve az általános (ld. 1.1.pont) függvény szerint. Az ott szereplő  $\alpha$  és  $\beta$  nemnegatív paraméterek beállíthatók, alapértelmezésük itt  $\alpha = 0.15$ ,  $\beta = 4.0$

Élenként az alábbi adatok adandók meg:

- éltípuskód (opcionális)
- a nulla forgalomnagysághoz tartozó utazási idő
- előtöltési forgalomnagyság (más ráterhelésből is vehető)
- forgalmi kapacitás

Az *éltípus* az alábbi adatokat jelenti:

- $\alpha, \beta$  paraméterek
- a 0 forgalomnagysághoz tartozó utazási sebesség
- élhossz
- sávonkénti kapacitás
- sávok száma
- költségszámítási konstansok (ld. alább)

A *csomóponti* ellenállásokat és tiltásokat a rendszer kezeli, ezek konstans táblázatok (csomóponti mátrixok) formájában adhatók meg. Ezek a táblázatok választhatók egy előre adott készletből, vagy közvetlenül is kitölthetők. A csomóponti ellenállás forgalomfüggetlen.

A forgalomráterhelés, mint standard output információt az élenkénti és csomópontonkénti *terhelést* és a *terhelt élidőt* adja. Ebből a rendszer más programjai további jellemzőket számolhatnak. Ilyen például a *környezetszennyezési mutató* amihez természetesen még más adatok is szükségesek.

Mint specialitást a rendszer a *kritikus élek* forgalmának gyűjtését kezeli. Ez a dolgozatunkban a 4.3.2. pont alatt ismertetett funkció egy részének tekinthető.

A ráterhelés történhet költség jellegű ellenállással is. Ez élköltséget jelent, a következő költségfüggvénnyel:

$$g = k + \omega_1 * t + \omega_2 * c$$

ahol  $\sigma$  az él költsége,  $k$  az élhez megadott általános költségkonstans,  $\omega_1$  az idő értéke konstans (globális konstans),  $\tau$  az élidő,  $\omega_2$  az élhez adott egyéb költségek szorzója (globális konstans),  $c$  az élhez adott egyéb költségek (pl. útdíj) konstansa.

A rendszer kezeli a több *forgalmi réteg* problémáját is, ekkor az él ellenállása az egyes rétegek forgalmából adódó ellenállások összege lesz. Az egyensúlyi állapot rétegenként értendő, vagyis az elvi optimum esetén minden réteg - a saját költsége szerint számítva - egyensúlyi állapotban lesz. A számítások itt is azonos (egységjármű/óra) dimenzióban történnek, ehhez meg kell adni rétegenként a költségfüggvényt és az átszámítási tényezőt.

A programcsomag a teljesség igényével lép fel, messzemenően *integrált univerzális* eszközt kíván a közlekedéstervező kezébe adni, lefedve a tervezési metodika teljes vertikumát. A fentebb ismertetett általános ráterhelési modulon kívül tartalmazza még a tervezésben használatos alábbi fő modulokat:

- *Forgalomkeltés*: aggregációs modell, regressziós modell, kiegyenlítés
- *Forgalomszétosztás, előrebecslés*: globális növekedési tényező módszere, átlagtényező (körzetenkénti egy növekedési tényező) módszer, Fratar (körzetenkénti két növekedési tényező) módszer, gravitációs módszer, K-tényező gravitációs módszer
- *Forgalommegosztás*: aggregációs modell, regressziós modell, logit modell
- *Tömegközlekedési forgalomráterhelés*
- *Körzetmátrix kiegyenlítés/előrebecslés forgalomszámlálási adatok alapján*. Ez céljában és alapelveiben megegyezik a dolgozatunkban *kalibráció* (5.3.5. pont) néven tárgyalt modullal.

### 6.3. EMME/2

Más közlekedéstervezési szoftvertermékekhez hasonlóan, az *EMME/2* programcsomag első verziói is felsőoktatási körökben keletkeztek (*University of Montreal*), majd fejlesztése és forgalmazása átkerült a versenyszférába. A rendszer gazdája az INRO Consultants Inc. Montreal cég [26].

Az *EMME/2* egy nagy méretű és komplexitású világszerte elterjedt közlekedéstervezési rendszer. Többféle számítógépes környezetben is implementált, a szokásos PC-DOS és Windows (3.1, 95, NT) mellett UNIX és VMS operációs rendszer alatt is futtatható. A kezelhető feladatméret maximum értékei: 2800 körzet, 22400 pont, 56000 él, a tömegközlekedést tekintve 2800 viszonylat és 70000 megálló.

A rendszer széleskörű adatmanipulációs lehetőségekkel bír, a hálózati karbantartó és lekérdező funkciók grafikusán is teljeskörűen támogatottak. A továbbfeldolgozás, illetve a tágabb térinformatikai környezetbe való beágyazás céljaira rendelkezik a CAD és GIS rendszerekbe történő adatexportálási funkciókkal is.

A programcsomagot 5 kontinensen, 48 országban több mint 500 szervezet használja [26]. A rendszergazda INRO rendszeresen szervez nemzetközi és regionális *felhasználói konferenciákat*, demonstrációkat, alkalmazási tanfolyamokat, tréningeket és egyre több országban rendelkezik helyi képviseléssel is. A kapcsolattartás egyik eszköze a rendszergazda által fenntartott nemzetközi hálózati információs (World Wide Web :WWW) hely is.



A magyarországi képviselőt a Kvantitás-Consulting Kft. Budapest látja el. Az ott kapott tájékoztatás szerint a programcsomag magyarországi alkalmazói (a legutóbbi jelentősebb alkalmazásokkal): UVATERV (országos közúthálózat modellezése), KTI Rt. (vidéki nagyvárosok közlekedési modellezése [71]) és a Közlekedés Kft.(budapesti közlekedési modell).

A forgalomráterhelési feladatra vonatkozóan a rövid rendszerismertető szerint:

- A rendszer a *determinisztikus egyéni optimum* módszerét ajánlja elsősorban, de más (sztohasztikus egyéni optimum és mindent vagy semmit) eljárások is lehetségesek.
- Az optimalizálás szempontja lehet az idő vagy a költség.
- Az idő ill. költségfüggvények a felhasználó által szabadon választhatók, pontosabban: a függvények egy a rendszerhez tartozó makrónyelvi mechanizmussal - bizonyos formai és tartalmi korlátok között - mint felhasználói adatok specifikálhatók.
- A rendszer kezeli a több *forgalmi réteg* problémáját is, az optimum egy globális egyensúlyi állapot lesz.
- Előre adott fix forgalom (előtöltés) alkalmazható.
- Többféle járulékos eredmény (pl. környezetszennyezési mutatók) számítható.
- Lehetséges többféle, az utak elemzésével számítható eredmény gyűjtése is.

A programcsomag lehetővé teszi, körzetenkénti bontásban bizonyos infrastruktúrális adatok nyilvántartását és kezelését. Ezekből *mátrixmanipulációs* (előrebecslés, kiegyensúlyozás, szimmetrizálás) módszerekkel állíthatjuk elő a ráterhelés input körzetszámításait. A rendszer tartalmazza a *tömegközlekedési ráterhelés* egy modelljét is.

### 6.4. TRANSURS

A *TRANSURS* modellrendszer és közlekedéstervezési programcsomag hazai termék [59]. A fejlesztés a hetvenes évek végén a Közlekedéstudományi Kutatóintézetben kezdődött, és alapvetően a *dr. Monigl János* vezette munkacsoporthoz kötődik. A korábbi modellverziók az URBANUS, BUDA-PC, TRANSWAY, neveken szerepeltek, a *TRANSURS* ezek továbbfejlesztésével jött létre. A rendszer gazdája a TRANSMAN Kft. Budapest.

A *TRANSURS* elsősorban *nagyvárosi alkalmazásokra* szánt modell és programrendszer. Ezen a területen viszont a *teljesség* igényével lép fel, és ezt magas szinten meg is valósítja. A rendszer röviden a következő főbb tulajdonságokkal jellemezhető [59]:

- A *közlekedési folyamatok* a területfejlesztési és felhasználási tervezésbe ágyazva, valamennyi városi és elővárosi közlekedési módra vonatkozóan, komplex módon kerülnek leképezésre.
- A *személyközlekedési igények* a különböző lakossági csoportok napi *helyváltoztatási láncainak* figyelembevételével egyéni, a hasznossági választási modellekkel kerülnek leírásra.
- A *teherforgalmi* igények az országos és nemzetközi forgalmi áramok figyelembevételével egyszerűbb, növekedési tényező modellekkel kerülnek leírásra.
- A *közúti hálózati* forgalom számítása a forgalmi körülmények által befolyásolt általánosított költségek alapján, az egyensúlyi feltételek elérésére törekedve történik.
- A *tömegközlekedési hálózati* forgalom számítása a viszonylati rendszer kapcsolódásai és a menetrend figyelembevételével az utazási idők és költségek alapján lehetséges.
- A *hatásértékeléshez* a forgalmi modellekhez csatlakozó hatásmodellek szolgálnak, így a *forgalmi körülményektől* és *mennyiségektől* függő mutatók (pl. üzemköltségek,

légszennyezés, zaj), valamint az *eljutási lehetőségeket* minősítő mutatók (pl. a belváros megközelíthetősége, környéki kapcsolatok).

A programcsomag PC-DOS szoftverkörnyezetben fejlesztett, C programozási nyelven, futtató környezete is a DOS operációs rendszer. A kezelhető feladatméret a futtató környezettől függ, a jelenlegi verzió maximum beállításai: 600 körzet, 12000 pont, 16000 szakasz, a tömegközlekedést tekintve 1500 viszonylat és 10000 megálló.

A programcsomag tartalmazza mind az egyéni mind a tömegközlekedési ráterhelési modellt, de a fejlesztési folyamatban - nyilván az alkalmazások többségének jellege (nagyvárosi tömegközlekedés) által meghatározóan - a tömegközlekedési rész került részletesebb, pontosabb kimunkálásra, ráterhelési szempontból nézve a modellben ez a domináns. A modellben helye van minden, a nagyvárosi közlekedésben előforduló közlekedési eszköznek, illetve utazási módnak a (vonattól a gyaloglásig terjedően 14 féle utazási módot kezel).

A rendszer hálózati modellje nagyon részletes közúthálózati és tömegközlekedési hálózatleíró adatbázist tartalmaz (nagyon sokféle szakaszadat, csomóponti forgalomszabályozási adatok részletes viszonylatleírások, megállóhelyek kialakítási jellemzői stb.). A hálózatleírás mellett - környezetvédelmi mutatók számításához - környezetvédelmi szempontú területjellemző adatokat (pl. a terület jellege, zajvédelmi létesítmények) kezel.

A programrendszer a tartalmi szempontok mellett nagy hangsúlyt helyez a bemenő adatok és az eredményadatok grafikus megjelenítésére és könnyű kezelésére. Az eredmények továbbfeldolgozhatóságát a CAD rendszerekbe történő adatexportálási funkciók is támogatják.

A programrendszert a rendszergazda TRANSMAN Kft. üzemelteti. Alkalmazásai többsége a nagyvárosi komplex közlekedésfejlesztési tervezéséhez kötődő. Utóbbi jelentősebb alkalmazásai: a Budapesti Közlekedési Vállalat (BKV) számára készített forgalmi modell [57] és a Budapest Közlekedési Szövetség modellezése [58].

A programcsomagban a forgalomkeltés, megosztás és szétosztás az egyének illetve csoportok napi helyváltoztatási láncainak elemzésére illetve az ezekkel együtt használt logit típusú, hasznossági számítási modellekre alapul.

A *ráterhelést* tekintve, a programcsomagban a nagyvárosi és városkörnyéki alkalmazás követelményeinek megfelelően többféle ráterhelési módszer is meg van valósítva. A rendszer egyrészt implementálja a *lépcsős ráterhelés* módszerét másrészt tartalmaz egy *determinisztikus egyéni optimum* típusú ráterhelési lehetőséget is. A hálózati elemek ellenállása *általánosított költségben van kifejezve*.

A rendszer kezeli a több rétegből álló forgalmat is, beleértve a rétegenként eltérő hálózati jellemzők (pl. forgalmi tiltások) és eltérő paraméterezésű ellenállás (idő) függvények megadhatóságát is.

Az *útvonalválasztásnál* opcionálisan maximum az első három legrövidebb utat számítja és kezeli, de ezek kiválasztásánál az útvonalak költségeinek összevetésén túlmenően egy további, logikai elemzést is végez: nem veszi figyelembe azokat az alternatív útvonalakat, amelyek az útvonal valamely részén "közelítenek" az útvonal kezdőpontjához.

A *tömegközlekedési* ráterhelés céljaira a rendszer egyik eszköze egy a szakaszhálózatra, az átlagos követési idő alapján történő (vagyis a közúti analógiára alapuló) ráterhelés. A másik, bizonyos mértékben már szimulációs jellegű eljárás a *menetrendi időpontok* alapján "időkövetéssel" történő ráterhelés, amely a tömegközlekedés pontosabb leképezésére ad lehetőséget.

### 6.5. TRIPS

A *TRIPS* programrendszer az angliai központú MVA világcég terméke, egy nagy méretű és komplexitású, világszerte elterjedt közlekedéstervezési rendszer. Többféle számítógépes környezetben is implementált, a szokásos személyi számítógépes (DOS és Windows) lehetőségek mellett közép és nagygépeken is futtatható UNIX operációs rendszer alatt. A kezelhető feladatméret alapján véve a számítógépes környezettől és az aktuálisan alkalmazott (megvásárolt) kiépítettségtől függ, de van néhány abszolút korlátozás is, pl. maximum 10000 körzet és 50000 pont adható meg a közúti közlekedési modellben.

A programcsomag jelenlegi alapverziója 1994.évi kibocsátású és a 7.0 számot viseli [60]. Egyes modulok - ilyenek például az interaktív grafikus adatkezelés programjai - későbbi fejlesztések ( 7.1 verziószám), megjegyzem, hogy a korábbi verziók csak alfanumerikus adatmegadással dolgoztak. A programcsomag egyik fő jellemzője éppen a nagyfokú modularitás. A szolgáltatások nem csak formai - értve ezalatt a futtató környezetből adódó vagy a szolgáltatás kényelmi szintjét meghatározó (pl. grafikus-nem grafikus) eltéréseket - hanem bizonyos esetekben lényegi, tartalmi szinten is "szét vannak szedve" (ilyen például a csomóponti mozgások modellezhetősége).

A rendszer, az összes modulját [60] figyelembe véve magas színvonalú szolgáltatást nyújt mind az adatmanipuláció és eredménymegjelenítés lehetőségeit, mind a közlekedéstervezési feladatok modellezhetőségét, megvalósíthatóságát illetően.

A programcsomag világszerte ismert és használt [60]. A rendszergazda MVA Systematica szervezi a felhasználói csoportokat, tapasztalatcsere és konzultációs fórumokat, alkalmazási tanfolyamokat. Világszerte regionális képviselőket tart fenn. A szervezés, kapcsolattartás és információszolgáltatás egyik eszköze a rendszergazda által fenntartott WWW hely is.

A rendszer magyarországi képviselőt a KTI Rt. látja el. A programcsomag a győri Széchenyi István Főiskolán és a Budapesti Műszaki Egyetemen üzemel, kizárólag oktatási célú felhasználásban. Más magyarországi felhasználóról, illetőleg más jellegű felhasználásról jelenleg nincs tudomásunk.

A programcsomagon belül a *ráterhelési* modulban implementált módszerek:

- *Mindent vagy semmit eljárás*
- *Lépcsős ráterhelés*
- *Determinisztikus egyéni optimum*
- *Sztohasztikus egyéni optimum* (a Burrell [62] vagy a Dial [19] módszerrel)
- Egy *iterációs* eljárás (az 1.3.2. pontban leírtakhoz hasonló)

A rendszer tartalmaz egy, az eddig ismertetett - alapjában véve statikus - modellektől eltérő elvű, erősen szimulációs jellegű (időfüggő ellenállások, csomóponti forgalom szimulációs modellezése stb.) *dinamikus ráterhelési* modul is.

A élköltség alapesetben időben van kifejezve, alapvetően az általános (ld. 1.1. pont) függvény szerint, de ez itt az általános esetnél tágabban, több változóval paraméterezhető.

*Élenként* az alábbi adatok adandók meg:

- éltípus
- a nulla forgalomnagysághoz tartozó utazási sebesség vagy idő
- forgalmi kapacitás
- jellegkód

Az éltípus az él közlekedési fizikai jellemzőit kódolja (útkategória jellegű adat). A jellegkód részhálózatok definiálásának és az ezekre vonatkozó eredmények képzésének a lehetőségét tartalmazza (a 4.3. pontban ismertetethez hasonló szolgáltatások). Az élenkénti előtöltés lehetősége adott a ráterhelési modulban.

A rendszer sokféle *csomóponttípust* ismer és ezekhez tágan paraméterezhető ellenállásfüggvényeket rendel. Lehetővé teszi (a más rendszerekben csak felbontással modellezhető) olyan összetett csomópontok mint pl. a körforgalmi csomópont közvetlen definiálását és paraméterezését is. A *csomóponti* ellenállásokat és tiltásokat kezeli, ezek is nagyon széleskörűen paraméterezhetők. Lehetséges irányonkénti csomóponti kapacitás megadása is. A csomópontokhoz megadandó adatok jellege és részletezettsége erősen függ az alkalmazni kívánt ráterhelési eljárástól, a legtöbb csomópontleíró adatot természetesen a dinamikus ráterhelés igényli.

A ráterhelés útvonalválasztása költség jellegű ellenállásra is alapozható. Ennek egyik összetevője az élköltsége, a következő költségfüggvénnyel:

$$g = \omega_1 * t + \omega_2 * l + \omega_3 * c$$

másik komponense a csomóponti késleltetés költsége, amely  $\omega_1 * t$  alakú.

A képletekben:  $g$  az él költsége,  $\omega_1$  az idő értéke,  $t$  az idő,  $\omega_2$  az élhosszal arányos költségeket kifejező szorzó,  $l$  az él hossza,  $\omega_3$  az élhez adott egyéb költségek szorzója, a  $c$  pedig az élhez adott egyéb költségeket (pl. útdíj) jelenti. A  $\omega_1$ ,  $\omega_2$ ,  $\omega_3$  és  $c$  faktorok globálisan vagy él illetve ponttípusonkénti bontásban adhatók meg.

A forgalomráterhelés alapvető eredményei az élenkénti és csomópontonkénti (irányonként is bontott) *terhelés*, valamint az ezekre vonatkozó *terhelt idő* és *költségadatok*. Ebből a rendszer más programjai a széles skálában számolnak ki leszármaztatott jellemzőket.

A rendszer jellegkódonként külön gyűjti és elemzi az így megjelölt *kritikus élek* illetve élhalmazok eredményadatait (a 4.3. ponthoz hasonló szolgáltatások). Ugyancsak megtalálható a kijelölt *részhálózatokra* vonatkozó eredmények gyűjtése is (4.5. ponthoz hasonló szolgáltatás).

A rendszer a több *forgalmi réteg* problémáját a mátrixok azonos (egységjármű/óra) dimenzióra való átszámításával kezeli.

A rendszer egyik fő összetevője a részletesen kidolgozott *tömegközlekedési* ráterhelési modul. A programcsomag a tervezési metodika többi fázisaira is ad a rendszerbe integrált programmodulokat:

- *Forgalomkeltés*: aggregációs modell, regressziós modell, kiegyenlítés
- *Forgalomszétosztás, előrebecslés*: növekedési tényezős módszerek, gravitációs módszerek
- *Forgalommegosztás*: logit modell

A rendszer a forgalmi adatok kezelésére a *mátrixmanipulációs* programok széles választékát nyújtja.

### 6.6. Összevetés

A dolgozat jelen fejezetében négy rendszerrel foglalkoztunk. Hozzájuk véve a NETWINFO programcsomagot le is fedtük (a gyakorlatban is alkalmazható rendszereket tekintve) a magyarországi alkalmazási és oktatási választékot. Ez lehetőséget ad arra, hogy a szakmai momentumok szintjénél általánosabb szempontból nézve összevessük az ilyen rendszereket.

A tárgyalt rendszerek közül kettő hazai fejlesztés, három külföldi, illetve nemzetközi cégek terméke. Nevezzük röviden az előbbieket "hazai" az utóbbiakat "külföldi" rendszereknek vagy programoknak.

A külföldi programcsomagokról elmondható, hogy rendelkeznek mindazon tulajdonságokkal, amelyek a *modern nemzetközi szoftverpiacon* forgalmazott árucikkeket jellemzik:

- Többfajta, számítástechnikai szempontból egymástól lényegesen különböző *számítógépes környezetben* is implementálhatók
- Szakmai, alkalmazási szempontból is a *modularitás* jellemző rájuk. Ez egyrészt horizontális jellegű, tehát kínálati szinten igyekeznek lefedni a teljes tervezési igényt, sőt egyes feladatokra több megoldási lehetőséget tartalmaznak, másrészt vertikális jellegű, tehát ugyanannak a részfeladatnak több, eltérő részletezettségű-pontosságú-megjelenítési színvonalú megoldó programja van a rendszerben. Természetesen az akár mennyiségben, akár minőségileg eltérő szolgáltatási szintekhez eltérő árak is tartoznak.
- A fejlesztő/forgalmazó cégek aktív *marketing tevékenységet* folytatnak. Egyre több régióban és országban rendelkeznek helyi képvisellel is. A jogosult felhasználókat számontartják, konzultációs lehetőséget biztosítanak, kiképző-továbbképző tanfolyamokat kínálnak számukra. Jellemző, hogy a fentebb tárgyalt három külföldi rendszer mindegyike rendelkezik a - napjainkban egyre többet és hatékonyabban használt - INTERNET -es információszerzés/információcsere lehetőségével.
- Ezeket a rendszereket számos országban használják, egyeseknek közülük jó esélyük van arra, hogy mintegy tervezési szabvánnyá váljanak.

A hazai programcsomagok, más hazai szellemi termékekhez hasonlóan szintén magukon hordozzák a szülő gazdasági-társadalmi környezet jegyeit, aminek idevonatkozó jellemzői a tőkehiány, a viszonylag olcsó munkaerő és (a legutóbbi évekig) a mindenkori modern számítástechnikai eszközöktől való elzártság voltak. Ennek következtében:

- A hazai rendszerek kevésbé árujellegűek, inkább helyhez, felhasználóhoz, számítógépes platformhoz kötött, saját célra kifejlesztett *egyedi termelőeszközök*. Jellemző, hogy a felhasználói kör szűk, esetleg egyedül maga a rendszergazda az egyetlen felhasználó is.
- Ezeknek a programcsomagoknak a fejlesztése állandó *számítástechnikai erőforrás-hiányos* környezetben folyt (köznapián kifejezve: a feladat mindig nagyobb volt, mint amekkorát a rendelkezésre álló számítógépre egyszerűen rá lehetett volna vinni), ezért ugyanazon vagy hasonló színvonalú szolgáltatást adó program előállításához sokkal több emberi munkát igényelt mint a külföldi rendszereknél. Megjegyzem, hogy ennek van pozitív következménye is: a hazai programok sokkal "tömörebbek", jobb hatásfokúak (ugyanazt az eredményt kevesebb ráfordítással adják).

Mi volt, vagy mi lehet esetleg még ma is egy hazai rendszer előnye, ami a fejlesztést és alkalmazást indokolttá teszi a hazai felhasználó számára? Véleményünk szerint előny az, hogy:

- A fejlesztő általában elérhető, a módosítási, továbbfejlesztési igények viszonylag gyorsan és olcsón realizálhatók.
- A felhasználó egyéni ötletei, elképzelései rugalmasan bevezethetők a rendszerbe, ennek következtében az eredmények többfélék, erősebben specializáltak lehetnek.
- A rendszer betanulása rövidebb időt és kevesebb munkát igényel, egyrészt azért, mert számíthatunk a fejlesztők személyes támogatására, másrészt azért mert ezek a rendszerek egyszerűbbek, jobban a tervezési feladatra koncentrálnak, és kevesebb ehhez csak gyengén kötődő, inkább a számítástechnikai környezetből adódó elemet tartalmaznak.
- Ugyanazon vagy hasonló színvonalú eredményt, szolgáltatást - igaz, hogy esetleg több emberi munka (az eredmények utófeldolgozása, "csomagolása") árán - gyengébb számítástechnikai bázison is el lehet érni.

Természetesen tudjuk, hogy ezek az előnyök viszonylagosak. A közlekedéstervezési munka is egyre nyitottabbá, nemzetközibbé válik, elképzelhető, hogy néhány év múlva szabványszerűen kötelező lesz, vagy legalábbis megrendelői előírássá illetőleg lényeges előnnyel járó versenytárgyalási opcióvá válik valamely nemzetközileg is ismert rendszer használata. Erre való utalás, illetve az ilyen rendszerek kontroll jellegű használata már a mai tervezési gyakorlatban (koncessziós autópályák) is előfordul. Mint ez más területeken is várható, a hazai kutató-fejlesztő munka az elméleti-oktatási tevékenységre és a kész rendszerek adaptálására, esetleges kiegészítésére fog korlátozódni.

## Függelék

### 1. Gráfelméleti alapfogalmak

*Gráfon* bizonyos objektumok és a köztük fennálló *közvetlen* kapcsolatok halmazát értjük. Két objektum között vagy van vagy nincs *közvetlen* kapcsolat. Az objektumok és a kapcsolatok további jellemzői itt közömbösek. Nevezzük az objektumokat *pontoknak*, a kapcsolatokat *éleknek*. Grafikus ábrázolásban a *pontokat* egyszerű geometriai alakzatokkal (kör, négyzet ...), az *éleket* a pontokat összekötő szakaszokkal szoktuk jelölni (46. ábra). Például: Egy város utcahálózatának *kereszteződései* legyenek a *pontok*, az *összekötő* szakaszok az *élek*. Ilyen gráf lehet pl. a 46/a. ábra.

*Egyszerű gráf* az olyan gráf, amelyben bármely két pont között legfeljebb egy él van és nincs hurokél (hurokél: egy pont önmagával való kapcsolata pl. egy olyan útszakasz, amelynek kezdő és végpontja azonos). Mi itt csak egyszerű gráfokkal foglalkozunk, a közlekedési hálózatok ilyenekkel modellezhetők.

*Teljes gráf* az olyan gráf, amelyben bármely két pont között van él. Ez a tulajdonság a közlekedési hálózatokra nem jellemző.

*Út* a gráf egy olyan pont illetve élsorozata, amelynek a felsorolás (bejárás) sorrendjében szomszédos pontjai közt van él. (46/a. ábrán a vastag vonal). A *körös út* olyan út, amelyben van legalább egy olyan pont, amely ismétlődik. *Körmentes út* az, amelyben ilyen pont nincs.

*Fa struktúrájú gráf*, röviden *fa* az olyan gráf, amelyben körút csak élismétléssel hozható létre (46/b ábra).

*Összefüggő* a gráf, ha bármely két pont között van út. A közlekedési hálózatok modelljei a valóságnak megfelelően természetesen összefüggők.

*Írányított gráf* az olyan gráf, amelyben az élekhez irányítást is rendelünk, kifejezve ezzel a nem feltétlenül szimmetrikus kapcsolatokat. Az él irányítása nyilvánvalóan éppen háromféle lehet, egy **a** és egy **b** pont viszonylatában:

- az **a** -ből a **b** -be mutató (csak az **a-b** kapcsolat áll fenn)
- a **b** -ből az **a**-ba mutató (csak a **b-a** kapcsolat áll fenn)
- mindkét irányban mutató (mind az **a-b**, mind a **b-a** kapcsolat fennáll)

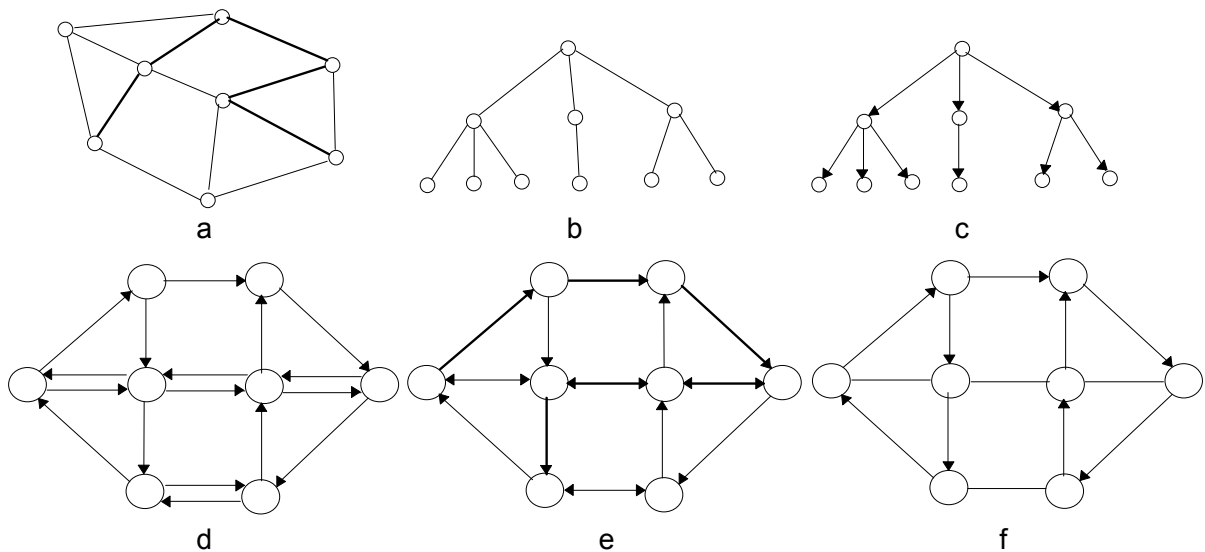
Az **x** pontból az **y** pontba mutató él jelölésére az **(x, y)** -t fogom használni.

A közlekedési hálózatok modelljei (a legegyszerűbb esetektől eltekintve) irányított gráfok a forgalomszabályozás ( egyirányú útszakaszok, nem szimmetrikus befordulási lehetőségek) modellezhetősége végett.

Az irányítást rajzban nyilakkal jelöljük (46/e. ábra). Sok esetben egyszerűbb a szimmetrikus (mindkét irányú) kapcsolatokat az irányítás nélkül jelölni (46/f. ábra), máskor viszont kifejezőbb (gondoljunk például egy utcaszakasz két oldalára) a *kétirányú* él elnevezés helyett *két darab egyirányú* élről beszélni és a rajzos ábrázolásmódban is ezt követni (46/d. ábra).

Az *irányított út*, *körös út* és *körmentes út*, valamint az *összefüggőség* az alapeset értelemszerű módosításával definiálhatók. Az út pontsorában a felsorolás (bejárás) sorrendjében egymás után következő pontjai közt a megelőzőből a következőbe mutató élnek kell léteznie (46/e. ábrán a vastag vonal).

*Irányított fa* az olyan irányított gráf, amelyben csak egyirányú élek vannak, és kijelölhető egy olyan pont, a *gyökérpont*, amelybe nem mutat él, viszont belőle az összes többi pontba pontosan egy irányított út vezet, más szóval minden ilyen pontba csak egy darab másik pontból mutathat él (46/c ábra). Ez a struktúra egy egyértelmű alá-fölérendeltségi vagy *előd-utód* viszonyrendszert definiál. Azokat a pontokat, amelyekből nincs kimutató él (nincs utódjuk) a *fa leveleinek* nevezzük. Az *irányított fák* a forgalomráterhelési számítástechnikai modellek fontos elemei, az útvonalválasztási algoritmusokban jutnak szerephez.



46. ábra. Gráfok

A modellezendő problémák jelentős részében - így a ráterhelésnél is - nem csak a pontok közti kapcsolatok megléte vagy hiánya fontos, hanem az is, hogy a kapcsolathoz egy *mennyiség*, mérőszám is tartozzon, például egy közlekedési hálózat **a**-ból **b**-be mutató éle milyen hosszú, vagy mennyi idő alatt jutunk el az élen haladva az egyik pontból a másikba, vagy mennyi a fajlagos utazási költség. Ha a modellünket ezzel bővítjük, vagyis az élekhez mérő, súlyozó számokat rendelünk, akkor már *hálózatról* beszélünk. Irányított gráf esetén ezek a számok irányonként is különbözhetnek, tehát egy kétirányú él két irányához tartozhat két különböző érték. A közlekedési alkalmazásokban általában nem csak az élekhez, hanem a pontokhoz is tartoznak ilyen adatok, például egy csomópont esetén a térképi koordináták vagy a forgalomirányítás módja.

A *matematikai hálózat* tehát a kapcsolatokat leíró *gráf* valamint a pontokhoz és az élekhez tartozó adatok, a *pontjellemzők* és az *éljellemzők* együttese.

*Irányított hálózatnak* nevezzük az irányított gráf alapú hálózatot. A közlekedési hálózat modellje általában ilyen. A kétirányú élek jellemzői irányonként különbözhetnek, kizárólag irányított utakkal dolgozunk, az eredményadatok is irányonkénti bontásban keletkeznek. Ezért a *kétirányú él* helyett a *két (irányított, egyirányú) élről* lehet beszélni, és ezt a terminológiát használom a dolgozatban is.



## 2. Számítógépes tesztelési eredmények

### 2.1. Minimális fák

#### Hálózatok

Az alábbi hálózatok tényleges forgalomráterhelési munkákban szerepelt tervezési szintű hálózatok.

- **GY\_TER**: Győr város egy vázlatos hálózata (összhossza 200 km).
- **GY\_DET**: Győr város egy részletesebb hálózata (összhossza 550 km).
- **HU\_NY**: Dunántúl egy hálózata (összhossza 6000 km).
- **HU\_22**: Magyarország egy hálózata (összhossza 22000 km).

A közlekedési hálózatok topológiailag a négyzetrács jellegű hálózatokra hasonlítanak. Ezért néhány ilyen hálózattal is teszteltünk. Ezeknél az élhosszakokat az  $[1,10]$  intervallumban egyenletes eloszlású véletlenszám-generátorral állítottuk elő. Ilyen hálózatok a **G2015**:  $20 \times 15$  méretű hálózat, **G3030**:  $30 \times 30$  méretű hálózat, **G5050**:  $50 \times 50$  méretű hálózat.

#### Algoritmusok

- **EA** : Alapeljárás az aktivitás halmazt ténylegesen halmazzal ( $\alpha$  típus) reprezentálva.
- **UA** : Új eljárás az **EA** -val megegyező halmazreprezentációval.
- **EB** : Alapeljárás az aktivitás halmazt a távolság szerint növekvően rendezett listával ( $\beta$  típus) reprezentálva.
- **UB**: Új eljárás az **EB** -vel megegyező halmazreprezentációval.
- **EC**: Alapeljárás az aktivitás halmazt a távolság szerinti kupac szerkezetű fával ( $\gamma$  típus) reprezentálva.
- **UC**: Új eljárás az **EC** -vel megegyező halmazreprezentációval.

A hálózatok minden kezdőpontjára előállítottuk a minimális fákat mindegyik algoritmussal. A kapott átlageredményeket az 1. táblázat adja meg. A táblázat sorainak részletes értelmezése:

- **Halmaz**: Az *aktivitás halmazok* átlagos elemszáma pontonként (új kész pont kiválasztásánál).
- **Lépés**: Egy fa meghatározásához szükséges *minimumkiválasztási ill. besorolási és átsorolási lépések* átlagos száma.
- **%** : Az új eljárás számadata az eredeti százalékában.

A hatékonyság javulásának leginkább objektív mértéke az aktivitás halmaz méretének csökkenése (a táblázatban a **Halmaz%** sor), ami a példahálózatoknál, mint a táblázatban látható 15-20 % -os mértékű. A halmazkezelés módjától függő mértékben ez a javulás jelentkezik áttételesen a besorolási és átsorolási lépésszám csökkenésében.

A táblázatban szándékosan nem futási idő értékeket közlünk. A *futási idő*, azonos szoftver-hardver környezet mellett is nagyon függ az algoritmus beprogramozási, kódolási módjától. Elvben jobb algoritmus, egy rosszabb hatásfokú kódolás mellett adhat gyengébb időeredményeket mint egy elvben rosszabb algoritmus egy jobb kódolás mellett. Gyakorlati

tapasztalatok alapján az állítható, hogy azonos szintű, minőségű beprogramozás esetén a futási idők javulási aránya közelítőleg megegyezik a táblázat Lépés% soraiban megadott arányokkal.

Hálózat	GY_TER	HU_NY	GY_DET	HU_22	G2015	G3030	G5050
Pont	301	926	1851	2662	300	900	2500
Él	796	2178	5114	7608	1130	3480	9800
El/Pont	2.64	2.35	2.76	2.86	3.77	3.87	3.92
EA, EB, EC Halmaz	15.46	23.03	37.56	48.41	21.72	35.38	59.88
UA, UB, UC Halmaz	12.71	19.52	30.54	39.28	17.48	28.17	48.53
<b>Halmaz %</b>	<b>82</b>	<b>85</b>	<b>81</b>	<b>81</b>	<b>80</b>	<b>80</b>	<b>81</b>
EA Lépés	4653	21324	69448	128863	6517	31838	149708
UA Lépés	3826	18083	55916	104554	5245	25356	121327
<b>UA Lépés %</b>	<b>82</b>	<b>85</b>	<b>81</b>	<b>81</b>	<b>80</b>	<b>80</b>	<b>81</b>
EB Lépés	1558	7323	28554	52092	1780	9167	44433
UB Lépés	1403	6684	25066	45558	1570	8331	40568
<b>UB Lépés %</b>	<b>90</b>	<b>91</b>	<b>87</b>	<b>87</b>	<b>88</b>	<b>90</b>	<b>91</b>
EC Lépés	1568	5284	11882	18411	1708	5757	17901
UC Lépés	1440	4978	10891	17005	1580	5304	16404
<b>UC Lépés %</b>	<b>92</b>	<b>94</b>	<b>92</b>	<b>92</b>	<b>92</b>	<b>92</b>	<b>92</b>

1. táblázat

## 2.2. Második utak

A tesztfutásokat az előző pontban már ismertetett valós tervezési hálózatokon végeztük. Az eredmények jelölései:

- **Iterációk száma:** A kezdőpontként szükséges iterációk átlagos száma a második utak számításánál
- **Körös %:** A körös utak száma az összes viszonylatok százalékában.

Hálózat	GY_TER	HU_NY	GY_DET	HU_22
Pont	301	926	1851	2662
Él	796	2178	5114	7608
<b>Iterációk száma</b>	<b>4.12</b>	<b>13.52</b>	<b>24.01</b>	<b>25.32</b>
<b>Körös %</b>	<b>0.00</b>	<b>0.01</b>	<b>0.05</b>	<b>0.06</b>

2. táblázat

3. Alkalmazási referenciák

	Téma	Alkalmazó	Időszak
1	Budapest tömegközlekedési fejlesztési terve	METROBER Budapest	1978 -1981
2	Győr tömegközlekedési fejlesztési terve	VÁTI Budapest	1981 - 1982
3	Győr általános rendezési terve	VÁTI, Budapest	1982
4	Eger, Kaposvár, Nyíregyháza tömegközlekedési fejlesztési terve	VÁTI, Budapest	1982 - 1984
5	Szeged, Nagykánizsa, Békéscsaba tömegközlekedési fejlesztési terve	Utilitás, Győr	1982 - 1986
6	A szegedi kerékpároshálózat fejlesztési terve	Utilitás, Győr	1984
7	Traffic survey and data elaboration on the Motorway M1	Bouygues(France) - Bauconsult (Hungary, Győr)	1992
8	Traffic survey and elaboration of traffic data for evaluating the tender of Expressway No. 9. and Danube bridge at Szekszárd	BAMEST (Italy) - Bauconsult (Hungary, Győr)	1992
9	Ergänzende Verkehrsprognose für die Grenzüberschreitenden Strassen	Ingenieurbüro Kribernegg (Österreich, Graz)	1992
10	Keszthely városközpont ideiglenes északi elkerülésének várható forgalmi hatása	KÖHÉM KIG Zalaegerszeg - Bauconsult Győr	1992
11	A 85. sz. út forgalmi vizsgálata	KÖHÉM KIG Győr - Bauconsult Győr	1992
12	A 82-83. sz. főutak közös győri bevezető szakaszán várható forgalmi terhelések	KÖHÉM KIG Győr - Bauconsult Győr	1992
13	Verkehrskonzept Meran, Gleisdorf und Badgastein	Ingenieurbüro Kribernegg Österreich, Graz	1993
14	Preliminary traffic data and forecasting on the Motorway M7 Hungary	BAMEST (Italy) - Bauconsult (Hungary, Győr)	1993
15	Traffic survey, forecast and sensitivity analysis of tolls for evaluating the tender of Motorway M5 on Hungary	Bouygues(France) - Bauconsult (Hungary, Győr)	1993 - 1994
16	Folyamatos forgalmi adatszolgáltatás és konzultáció Győr város úthálózatán az M1 autópálya ütemezett üzembehelyezésekor várható forgalmi változásokról	Polgármesteri Hivatal Győr - Bauconsult Győr	1993 - 1994
17	Verkehrsuntersuchung Mürzzuschlag	IKK Graz	1995
18	Az M3 autópálya forgalmi és díjbevételi tanulmánya	Object Kft. Budapest - Bauconsult Győr	1996
19	Verkehrsuntersuchung "Umfahrung Freistadt"	IKK Graz	1996
20	Verkehrsuntersuchung Halbanschlußstelle Gratkorn	IKK Graz	1996

21	Az M1 autópálya és a 10 sz. főút közötti forgalommegosztás vizsgálata	KTI Rt. - Bauconsult Győr	1996
	<b>Téma</b>	<b>Alkalmazó</b>	<b>Időszak</b>
22	Az M3 autópálya költség - haszon elemzéséhez szükséges forgalmi adatok előrebecslése a kísérő hálózaton	Trafficon Kft. Budapest - Bauconsult Győr	1996
23	A győri keleti (81 -es főút) elkerülő út várható forgalmának előrebecslése	Polgármesteri Hivatal Győr - Bauconsult Győr	1996
24	A magyarországi gyorsforgalmi úthálózaton lebonyolódó utazások rendszerességének és legfontosabb szokásjellemzőinek meghatározása	Trafficon Kft. Budapest - Bauconsult Győr	1997
25	Az M3 autópálya díjkezdvezmény rendszerének forgalmi és bevételi hatásvizsgálata	Trafficon Kft. Budapest - Bauconsult Győr	1997
26	Verkehrsuntersuchung Vollausbau Anschlußstelle Spielfeld	IKK Graz	1997
27	Az M7 autópálya forgalmi vizsgálata	Bauconsult Győr	1997
28	Verkehrsuntersuchung 1423 Münzbacher Straße, Baulos "Zubringer Münzbach"	IKK Graz	1997
29	Az M3 korridor úthasználati jellemzőinek vizsgálata	Trafficon Kft. Budapest - Bauconsult Győr	1997
30	Az országos gyorsforgalmi hálózat fejlesztésének tervezéséhez szükséges kalibrált áramlási mátrixok meghatározása	UKIG Budapest - Bauconsult Győr	1997
31	Az M7 autópálya forgalmi méretezésének eredményei és fejlesztésének javasolt ütemezése	ENCON Kft. Budapest - Bauconsult Győr	1998
32	Az M30 autópálya korridor forgalmi vizsgálata, forgalmi és díjbevételei prognózisai	Trafficon Kft. Budapest - Bauconsult Győr	1998
33	Az M9 autópálya déli szektorának forgalmi és díjbevételei prognózisai	UKIG Budapest - Bauconsult Győr	1998
34	Az M43 autópálya forgalmi és díjbevételei prognózisai	UKIG Budapest - Bauconsult Győr	1998
35	A szekszárdi Duna - híd díjérzékenységi vizsgálatai	UKIG Budapest - Bauconsult Győr	1998
36	Az M5 korridor hatásterületének forgalmi vizsgálatai valamint forgalmi és díjbevételei prognózisai	UKIG Budapest - Bauconsult Győr	1998
37	Verkehrsuntersuchung Umfahrung Hartberg	IKK Graz	1998

## Irodalomjegyzék

- [1] Aho, A.V. - Hopcroft, J.E. - Ullman, J.D.: *Számítógép algoritmusok tervezése és analízise*  
Műszaki Könyvkiadó Budapest 1982
- [2] Ábrahám, K. (főszerk.): *A közúti közlekedés kézikönyve*  
Műszaki Könyvkiadó Budapest 1984.
- [3] Bakó, A.: *Forgalom szétosztás és elosztás tervezési modelljeinek matematikai és számítástechnikai értékelése és fejlesztése*  
Kandidátusi értekezés Budapest 1983.
- [4] Bakó, A.: *Ritka mátrixok számítógépes tárolási módszerei*  
KTMF Tudományos Közlemények **2**, 351-357 (1976)
- [5] Bakó, A.-Marton, L.-Takács, B.-Vesztergál L.: *Egy interaktív modell úthálózatok optimalizálására.*  
KTMF Tudományos Közlemények **1**, 33-36 (1981)
- [6] Bakó, A.-Marton, L.-Takács, B.-Vesztergál, L.: *Városi úthálózat változatok gazdaságossági elemzése.*  
Városi Közlekedés **5**, 245-262 (1981)
- [7] Bakó, A.-Marton, L.: *AMT eljárás területi tervezési feladatok megoldására.*  
I. Magyar AMT Konferencia, Budapest, 1982.
- [8] Bakó, A.-Marton, L.: *Komplex forgalomtervezési eljárások számítógépes megvalósítása.*  
KTMF IV. Tudományos Ülésszak, Győr, KTMF 1984.
- [9] Bakó, A.- Marton, L.: *Road networks and traffic information system*  
Conference of computer applications in road and transport planning. Győr, KTMF 1990
- [10] Bakó, A.-Berényi, J.-Marton, L.-Szántai, T.: *Gyalogosforgalmi terek tervezése személyi számítógéppel.*  
KTMF V. Tudományos Ülésszak, Győr, KTMF 1987.
- [11] Braun, J. - Schaechterle, K.: *Vergleichende Untersuchung vorhandener Verfahren für Verkehrsumlegungen*  
Forschung Strassenbau und Strassenverkehrstechnik  
**222**, 45 - 50 (1977)
- [12] CALIPER Corporation: *TransCAD Transportation GIS Software User's Guide Version 3.0*  
Caliper Corporation USA Newton, Massachusetts (1996)
- [13] CALIPER Corporation: *Travel Demand Modeling with TransCAD*  
Caliper Corporation USA Newton, Massachusetts (1996)
- [14] Clarke, S. - Krikorian, R. - Rausen, J.: *Computing the N best loopless paths in a network*  
J. Soc. Indust. Appl. Math. **11**, 1096-1102 (1963)
- [15] Csicsely-Tarpay, M.-Bakó, A.-Gáspár, L.-Marton, L.: *Hungarian pavement management system for the road network of a city.*  
International Conference on Pavement and Airfield management.  
Singapur, 1995.
- [16] Denardo, E.V. - Fox, B.L.: *Shortest - route methods 1., 2.*  
Operations Research **27**, 161-186, 548-566 (1979)
- [17] Deo, N - Pang, C.: *Shortest-Path Algorithms: Taxonomy and Annotation*  
Networks **14**, 275-323 (1984)
- [18] Dial, R.B.: *Algorithm 360: Shortest path Forest with Topological Ordering*  
Comm. of ACM **12**, 632-633 (1969)

- [19] Dial, R.B.: *A probabilistic traffic assignment model which obviates path enumeration*  
Transportation Research **5**, 83-111 (1971)
- [20] Dijkstra, E.W.: *A Note on Two Problems in Connexion with Graphs*  
Numerische Mathematik **1**, 269 - 271 (1959)
- [21] Florian, M. (editor) *Transportation Planning Models*  
North - Holland Elsevier, Amsterdam 1984.
- [22] Frank, M.- Wolfe, P.: *An algorithm of Quadratic Programming*  
Naval Research Logistics Quarterly **3**, 95-110 (1956)
- [23] Furness, K.P.: *Time function iteration*  
Traffic Engineering and Control **7**, 458-460 (1965)
- [24] Gallo, G. - Pallottino, S: *Shortest path methods in transportation models*  
Transportation Planning Models, M. Florian (editor)  
North - Holland 1984.
- [25] Hoffman, W. - Pavley R.: *A method for the solution of the n-th best path problem*  
Journal of ACM **6**, 506-514 (1959)
- [26] INRO Consultants Inc. *EMME/2 Transportation Planning System*  
(rövid rendszermertető)  
INRO Consultants Inc. Montreal, Canada (1996)
- [27] Jansen, G.-Bovy, P.: *Ein belastungabhängiges Umlegungsverfahren: Theorie und Anwendung des Gleichgewichtsmodells*  
Straßenverkehrstechnik **1**, 17-25 (1983)
- [28] Katoh, N.-Ibaraki, T.-Mine, H.: *An efficient algorithm for K shortest simple paths*  
Networks **12**, 411-427 (1982)
- [29] Kálmán, L.-Koren, Cs.-Marton, L.: *Informationssystem für Strassennetze*  
Computeranwendungen für Strassenentwurf und Verkehrsplanung,  
Internationaler Workshop Technische Universität Graz, 1991
- [30] Kálmán, L.-Marton, L.-Pusztai, P.: *A NETWINFO úthálózati és forgalmi információs rendszer.*  
Közúti közlekedési nyári egyetem Győr, 1993
- [31] Kálmán, L.: *Célforgalmi mátrixok számítása a továbbfejlesztett mátrixkiegénylítő algoritmussal*  
Közlekedéstudományi Szemle **5**, 1987
- [32] Kálmán, L.: *Közlekedéstervezés*  
Szakmérnöki jegyzet, KTMF 1982.
- [33] Khisty, C.J.: *Transportation Engineering*  
Prentice Hall 1990.
- [34] Kirby, R.F. - Potts R.B: *The Minimum Route Problem for Network with Turn Penalties and Prohibitions*  
Transportation Res. **3**, 397-408 (1969)
- [35] Klafszky, E.: *Az Input - Output tábla előrebecsléséről.*  
MTA - SZTAKI Közlemények **10**, 1-13 (1973)
- [36] Koller, S.: *Forgalomtechnika és közlekedéstervezés*  
Műszaki Könyvkiadó Bp. 1986.
- [37] Koren, Cs.-Kálmán, L.: *Methoden der Verkehrs- und Einnahmensprognosen auf Mautautobahnen*  
Proc. Konf. MOBILITA Slovak Technical University,  
Bratislava 1995.

- [38] Koren,Cs.: *Forgalmi mátrixok kiegészítése és felújítása keresztmetszeti számlálások alapján*  
Városi Közlekedés **2**, 86-90 (1983)
- [39] Koren,Cs.- Kálmán,L.: *Az 1987.évi országos közúti célforgalmi vizsgálat*  
Közlekedéstudományi Szemle **9**, 391-396 (1989)
- [40] LeBlanc,L.-Morlok,E.-Pierskalla,W. *An efficient approach to solving the road network equilibrium traffic assignment problem*  
Transportation Science **9**, 209-218 (1975)
- [41] Loubal, P.S.: *A network evaluation procedure*  
Highway Res. Rec. **205**, 96-109 (1967)
- [42] Marton,L.- Bakó, A.: *A k-adik legrövidebb út meghatározása csomóponti veszteségek esetén.*  
KTMF Tudományos Közlemények **1**, 371-375 (1978)
- [43] Marton, L. - Zaupper, T.: *K-adik utas algoritmusok*  
Matematikai és számítástechnikai módszerek a közlekedés tervezésében és irányításában II. Győr, KTMF 26 - 32 (1978)
- [44] Marton L.: *Minimálisút algoritmusok közlekedési hálózatokra.*  
KTMF Tudományos Közlemények **2**, 219-222 (1979)
- [45] Marton,L.- Pusztai,P.:*Informationssystem und Verkehrsrumlegung für Strassennetze*  
Computeranwendungen für Strassenentwurf und  
Verkehrsplanung, Internationaler Workshop -Berichte  
Technische Universität Graz, 67-68 (1993)
- [46] Marton L.: *Számítógépes forgalomelosztás*  
Új Alaplap **6**, 7-10 (1996)
- [47] Marton L.: *A NETWINFO programrendszer*  
Új Alaplap **6**, 13-14 (1996)
- [48] Marton, L.: *A csomópontok kezelése a forgalomelosztási eljárásban*  
Városi Közlekedés **5**, 281-284 (1996)
- [49] Marton,L.: *A forgalomelosztási feladat egy útkereső eljárása.*  
Közlekedéstudományi Szemle **7**, 248-253 (1996)
- [50] Marton,L.: *Az autópálya viszonylati díjkezdvezmények modellezése a számítógépes forgalomelosztási eljárásban*  
Közúti Közlekedés- és Mélyépítéstudományi Szemle  
**11**, 415-417 (1997)
- [51] Marton L.- Pusztai,P.: *Gráfok és hálózatok kezelése számítógéppel I.-VI. (cikksorozat)*  
Új Alaplap **4-9**, (1997)
- [52] Marton,L.: *Egy címkéző eljárás a legrövidebb utak fájának meghatározására ritka hálózatokban*  
Alkalmazott Matematikai Lapok **19** (1999) (közlésre elfogadva)
- [53] Marton,L.: *Modelling motorway toll in traffic assignment*  
Slovak Journal of Civil Engineering (1999) (közlésre elfogadva)
- [54] Marton,L.- Pusztai,P.:*A NETWINFO programrendszer felhasználói kézikönyve*  
Bauconsult Mérnökiroda, Győr (1998)
- [55] Marton,L.: *On Modelling Traffic Assignment*  
Hungarian Electronic Journal, <http://heja.szif.hu>  
(közlésre leadva)
- [56] Marton,L.: *A label-setting method to determine the tree of shortest paths in sparse networks*  
Central European Journal of Operations Research (közlésre leadva)

- [57] Monigl,J. - Koren, T. - Újhelyi,Z.: *BKV forgalmi modell*  
TRANSMAN Kft. Budapest tanulmány (1995)
- [58] Monigl,J. - Újhelyi,Z. - Koren, T. - Berki, Zs. - Nagy, E. : *A Budapesti Közlekedési Szövetség (BKSz) megalapozó vizsgálata*  
Városi Közlekedés **4**, 209 -228 (1997)
- [59] Monigl,J.:  
*TRANSURS modellrendszer*  
TRANSMAN Kft. Budapest tanulmány (1998)
- [60] MVA Systematica: *TRIPS manuals (Version Seven)*  
MVA Systematica, 1994.
- [61] Nagy,E. - Szabó,D. (szerk.) : *Városi közlekedési kézikönyv*  
Műszaki Könyvkiadó Bp. 1984.
- [62] Ortúzar,J.D.-Willumsen,L.G.: *Modelling Transport*  
John Wiley & Sons1994
- [63] Perko, A.:  
*Implementation of Algorithms for K shortest loopless paths*  
Networks **16**, 149-160 (1986)
- [64] Prilevszky,I.- Fülöp,G.: *Közlekedéstervezés.*  
Távoktatási jegyzet, Széchenyi István Főiskola Győr, 1996.
- [65] Schnabel,W. - Lohse,D.: *Grundlagen der Strassenverkehrstechnik und der Verkehrsplanung*  
Verlag für Bauwesen, Berlin 1997
- [66] Sheffi,Y. - Powell,W.: *An algorithm for the Equilibrium Assignment Problem with Random Link Times*  
Networks **12**, 191 - 207 (1982)
- [67] Sheffi,Y. - Powell,W. *A comparison of stochastic and deterministic traffic assignment over congested networks*  
Transportation Research-B **15B**, 53 - 64 (1981)
- [68] Shier, D.:  
*On algorithms for finding the k shortest paths in a network*  
Networks **9**, 195-214 (1979)
- [69] Steierwald,M. - Künne,H.G.: *Stadtverkehrsplanung*  
Springer Verlag 1994.
- [70] Transportation Research Board: *Highway Capacity Manual.*  
National Research Council Washington 1985 - 1997
- [71] Vörös, A.:  
*Települések közlekedési problémái*  
Önkormányzati ügyek '97 konferencia Tatabánya 1997.  
Esettanulmányok 72 -77.
- [72] Wardrop,J.G.:  
*Some theoretical aspects of road traffic research*  
Proceedings Institute of Civil Engineering **36**. (1952).
- [73] Warshall, S.:  
*A Theorem on Boolean Matrices*  
Journal of ACM **9**, 11-12 (1962)
- [74] Weigand, M.:  
*Algorithmus 27: Ein neuer Algorithmus zur Bestimmung von k-kürzesten Wegen in einem Graphen*  
Computing **16**, 139-151 (1976)
- [75] Yen, J.:  
*Finding the K shortest loopless paths in a network*  
Manag. Sci. **17**, 712-716 (1971)

**Megjegyzés:** A saját publikációk közül elektronikusan kereshetők és MS Word 6.0 dokumentum formájában letölthetők a 48,49,50,51,52,53,54 és 55 hivatkozási számúak. Az elérési cím: <http://rs1.szif.hu/~marton/publikal.htm>.