

Biztosítóberendezési függőségek számítógépes tervezése

© Székely Béla

Mit is értünk függőségi rendszer alatt?

Ha nagy általánosságban és jogi értelemben akarjuk megfogalmazni, elegendő azt mondanunk, hogy a felhasználó számára releváns adatok összessége. A hagyományos biztosítóberendezésekhez szokott gondolkodásmódunkban ez főleg a menettervi és elzárási adatokat jelenti. Tervezői szemmel azonban ezek a függőségi tervek „csak” szerződéses dokumentumok. A tényleges függőségi rendszer megvalósítása olyan aprólékos munka, amelynek alapja az adott biztosítóberendezés belső felhasználású tervezési segédletei. Ezek megjelenhetnek kilincsek, vonalzó, alapkapsolások vagy kapcsolási esetek formájában. Ha van tehát egy vágánygeometria, van egy jó alapkapsolás (tervezési utasítás), és minden egyéb szabály jól dokumentált, egy biztosítóberendezés gondolkodás nélkül megtervezhető volna. Csak ez a sok feltételes mód ne volna!

És az elektronika?

Ha az elektronika, számítástechnika ellen emelnék szót, egészen bizonyos, hogy nem a jövőbe mutató gondolkodásmódot képviselném. Nem kell azonban nagy bátorság annak kijelentéséhez, hogy az elektronikus biztosítóberendezések nem teljesítették maradéktalanul a kezdetekben hozzájuk fűzött reményeket.

- Az elektronikus berendezések olcsók lesznek.

Miközben a számítástechnikai elemek ára töredékére zuhant, az elektronikus/jelfogós biztosítóberendezések költséghányada nőtt.

- Az elektronikus berendezések gyorsan telepíthetők lesznek.

Volt rá eset, hogy a megrendeléstől számított három hónapon belül üzembe helyeztünk egy D55 berendezést.

- Az elektronikus berendezésekben nincs kapcsolóelem, élettartamuk korlátlan.

Ma már átépítjük az első elektronikus berendezéseket.

- Az elektronikus berendezések energiatakarékosak.

Nem, energiafelhasználásuk lényegesen magasabb.

- Az elektronikus berendezések tervezése leegyszerűsödik.
Ezt kifejteném!

Önmagam ellenőrzésére taláломra elővettem egy alapkapsolást, és véletlenszerűen rámutattam érintkezőkre. Úgy gondolom, több százan jutottak volna hasonló eredményre, miszerint nagy biztonsággal meg tudtam mondani a taláломra kiválasztott érintkezők funkcióját.

Ezt követően elővettem egy általam megírt (biztosítóberendezési funkciót megvalósító) programot. Itt az alapvető funkciók megértése is csak nagy nehézségek árán sikerült. Ezzel – gondolom – mások is így vannak, de amíg jelfogós berendezések esetén több száz potenciális tervezőről, szakemberről beszélhetünk, elektronikus biztosítóberendezés szoftverével, vagy legalábbis annak környezetével foglalkozó szakemberek száma rendkívül csekély. Ebből azt a következtetést vonnám le, hogy az elektronikus biztosítóberendezések „termelése” nem egy olyan szakmatársadalmi folyamat, amelynek fejlődését a „hangyák” sokasága generálja. Az Integra egyközpontos berendezés fejlesztése, magyar szakemberek bevonásával történt. A D55 berendezés a svájci megalkotás után két-három évvel már Magyarországon is üzemelt, az Integra D67 – nálunk D70 néven – üzemelt Dunakeszin (a szám évszámra utal). Különösen a D55, kevésbé a D70 műszaki fejlődését magyar koponyák generálták. A már több mint tíz éve létező elektronikus rendszerek lelkiállagának megismerése még mindig a ködös jövő része a biztberes társadalom számára. Kétes értékű vigasz, hogy az elektronikus berendezések lelkét ismerő szakemberek száma a „nagy vasutak”-nál is csekély, és kétségeket ébreszt az, hogy az elektronikus berendezések gyártási mennyisége a korábbi évekhez képest jelentősen csökkent, ezzel együtt a fejlesztési kedv, összességében tehát a fejlesztéssel foglalkozó szakemberek száma is. Ennek ellenére, vagy pont ezért ez a folyamat a MÁV számára akár kedvező is lehet. A nagy gyártó cégek kezdik felismerni, a magyar szakemberek fejlesztési folyamatba történő bevonásának gazdasági és szakmai jelentőségét. A fenn vázolt problémák (részleges) megoldását csak az jelentheti, ha a tervezési, projektálási folyamat ideje, a folyamatban részt vevők száma radikálisan csökken. Ennek pedig előfeltétele a tervező eszközök fejlesztése.

Néhány gondolat a máig ható „távoli” múlttól

A számítástechnika hajnalán (amikor még a jobban hangzó kibernetika szót gyakrabban használtuk) kissé önfényezésen azt mondtuk, hogy a (jelfogós) biztosítóberendezés tulajdonképpen egy fix programos számítógép. Amikor a tranzistorok, kapuáramkörök kézzel fogható közelségbe kerültek, lelkesen kezdtünk olyan áramköröket gyártani, amelyekkel jelfogókat, jelfogó egységeket, funkciókat lehetett volna kiváltani. Születtek ugyan zseniális megoldások bizonyos hibafajták kiküszöbölésére, de alapvetően nem tudtunk mit kezdeni azal a ténnyel, hogy egy kapcsolóelem meghibásodhat. A hibát a rendszernek fel kell ismerni és biztonságosan leereálgálni. Ez lehűtötte lelkesedésünket. Meglepő módon hosszú időre bénította gondolkodásunkat a processzorok megjelenése. Ezek megismerése ugyanis hardver és szoftver szinten már igen jelentős szellemi befektetést, programozásuk, gyártásuk pedig elérhetően magas anyagi erőforrásokat igényelt. Ezek után a cél, már nem lehetett más, mint a megfelelő erőforrással rendelkező cégek felé közeledni.

Miért is olyan drága a biztonsági szoftver? Nem vagyok közgazdász, de talán nem tűnik hiteltelennek, ha azt mondom, hiányoznak mindazok a közgazdasági feltételek – nagyszámú fogyasztó, nagyszámú gyártó –, amelyek lenyomják az árakat. Műszaki szempontból is van magyarázat. A voltokon, ampereken és vezetékeken szocializálódott gondolkodásunk sokkal könnyebben feldolgozza a látható, jól strukturált áramköröket, mint a láthatatlan, és soha nem tanult folyamatokat. Egy profán hasonlattal élve táblán sakkozni könnyebb, mint vakon. A gondolkodás, és mások gondolatának megértése (ellenőrzések végett) pedig költséges dolog.

A kis példányszám miatt a szoftver, hardver tehát drága, a szoftveres nem ért az adott szakmához, a szakember pedig nem ért a szoftverekhez. Ezen okok miatt születtek olyan általános célú hardverek, amelyek tudnak áramot ki- és bekapcsolni (bűnös egyszerűsítés), illetve megszületett a szakértői programnyelv fogalma. Ez azt jelenti, hogy ennek segítségével pl. egy biztberes szakember meg tudja írni a biztosítóberendezés programját. Ez a programnyelv úgy néz ki, hogy az egyetlen bal oldalán fel vannak sorolva a feltételek. Ha ezek a feltételek teljesülnek, a halmaz „tüzel”, azaz végre hajtja az egyetlen jobb oldalán lévő műveletet. Nos, ez a gondolat nagyon szép, működik is, de a fordító programok soha nem jutottak el olyan szintre, hogy a szakem-

berek ülték volna le programozni. Születtek azonban olyan biztonsági berendezések (elvek), amelyek magja alkalmazható a repülésben, hajózásban, vasúti vagy közúti közlekedésben, azaz mindent, ahol biztonság szükséges. Milyen hatással volt ez a fenti elvek szerint megvalósított biztosítóberendezésre, illetve a cikk címétől szolgáló függőségi rendszerek tervezésére. A biztosítóberendezést működtető szoftverek írójának – ebben az esetben talán projektálónak kellene nevezni – valamilyen formában adatokat kell kapnia. Az adathalmaz pedig – mivel vannak feltételek, azaz oszlopok, és vannak feltétel halmazok, azaz sorok – célszerűen táblázatos formátumú. Az előtervi adatokat tehát csak egy nagyon egyszerű editorral kell a projektálónak a rendszerrel „megetetni”.

Mivel a feltételek és a feltétel halmazok száma az állomás méretével exponenciálisan nő, ezeknek a rendszereknek mérete gyakran korlátozott.

Az általános, szakértő szemléletű biztonsági rendszerek mellett hamar megjelentek a speciális célú szoftverek. (A hardverelemek, elsősorban a processzorkártyák még sokáig általános célúak maradnak). Ezek a berendezések, bár tartalmazhatnak általános célú modulokat, mint pl. biztonsági mechanizmusok, szavazógépek stb., de alapvetően mégiscsak biztosítóberendezési funkciókat, forgalomszabályozási szolgáltatásokat valósítanak meg. Nagy problémája ennek a gondolkodásmódnak, hogy bonyolult a szoftver, és a biztonsági igazolása is rendkívül körülményes. Ez a módszer ezért inkább divergens programozáson alapuló biztonsági mechanizmust igényel. A második csatorna inkább ellenőrző funkciókat valósít meg, és megvalósítása a szakértői gondolkodásmódot kell hogy kövesse.

Egy ilyen rendszer megvalósítása nyilvánvalóan lényegesen nagyobb, bonyolultabb feladat. Ennek az első pillanatra tehát lényegesen drágább szoftvernek mégis van egy hatalmas előnye. Bele lehet építeni a bevezetőben említett szabályrendszert. Ha ez sikerült, akkor nincs más hátra, mint egy jó editorral felépíteni az állomás topológiáját, és betölteni az adatokat a biztosítóberendezésbe. De mi történik akkor, ha nem tudunk minden szabályt algoritmizálni vagy speciális forgalmi igények lépnek fel? Egy kétcsatornás divergens program módosítása, különösen, ha az biztonsági mechanizmusokat is érint, akár évekig tartó huzavonát eredményezhet. A fentiekből következően a specializálódott rendszereket, bár sok táblázatos jellegű adatot igényelnek – a rendszer fontos részét képező editor funkcióit, szolgáltatásait is fi-

gyelembé véve – nyomvonalas elvűnek nevezhetjük.

A fent leírtak talán túlzóan általánosítók, és talán vitára is okot adnak, de megállapításai alapvetően igazak a Magyarországon működő két biztosítóberendezés típusra. A Siemens által gyártott SIMIS IS berendezés táblázatos elven működő, míg a Thales által gyártott ELEKTRA inkább nyomvonal elvű tervezést igényel.

Mint mind a két berendezéssel szoros kapcsolatba (és barátságba) került tervező/fejlesztő nem tudnék egyik vagy másik rendszer mellett lándzsát törni.

A táblázatos formának azonban van két lényeges hátránya.

Tervezőkor egy nagy elzárási táblázat készítése közben, vagy egy közelítő vágányút táblázatos leírásakor nagyon könnyű hibázni, a jóváhagyás során nehéz a hibát felfedezni.

Funkcionális vizsgálatkor a táblázat minden egyes elemét vizsgálni kell. Egy nyomvonalas berendezésnél az objektumon áthaladó nyom függőségét elegendő csak egy vágányútban megvizsgálni, és a rendszer biztosítja, hogy az minden, az objektumon áthaladó vágányútra érvényes.

A FELADAT

A Siemens elhatározta, hogy a jövőben az engedélyezési és előterveit a nyomvonalas logika szerint fogja szállítani. Mivel egy biztosítóberendezés (részleges) nyomvonalasítása nem történhet egyik percről (évről) a másikra, a projektáláshoz továbbra is szüksége van a függőség táblázatos leírásához. Az AXON-

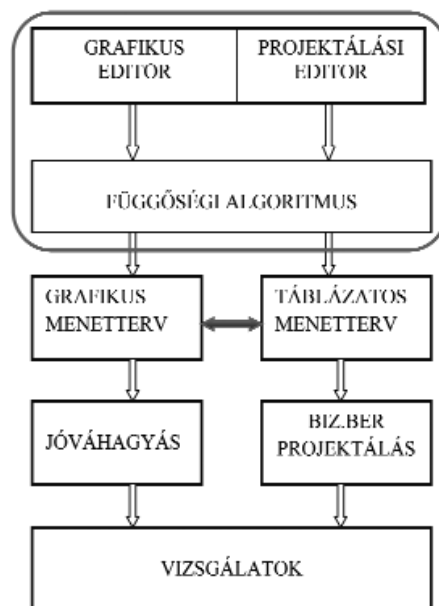
6M és a Bi-Logik Kft vállalta egy olyan tervező projektáló rendszer (tool) kifejlesztését, amelynek egyik kimenete generálja a grafikus függőségi tervet, ugyanakkor generálja a SIMIS IS számára szükséges függőségi táblázatokat is. Értelemszerűen biztosítani kell a két adathalmaz azonos információtartalmát, egyértelmű összerendelhetőségét.

A TERVEZŐ TOOL FUNKCIÓI

Egy biztosítóberendezés terve többféle stílusú dokumentumból áll. Nyilvánvalóan azokat a dokumentumokat, amelyek nem igényelnek logikai feldolgozást, mint pl. az ún. ezres rajz, ezt kereskedelmi CAD szoftverekkel kell tervezni. Ezek a szoftverek bár különböző mértékig nyitottak, csak korlátozott mértékig alkalmaznak függőségeket is tartalmazó adathalmaz létrehozására. Ha tehát biztosítóberendezések függőségi rendszerének tervezéséről beszélünk, elengedhetetlen egy speciális editor megírása.

Az editorral meg kell tudni szerkeszteni a vágányhálózaton alapuló függőségi terv vázát. Az editor másik kimenete az állomást nem csak grafikusán, hanem matematikai értelemben is leíró adatbázis. Ez az utóbbi szolgál bemenetül a függőséget generáló modul számára. Az editor grafikus felületének – összhangban az állomás leírásával (lerajzolásával) – meg kell tudni jelenítenie a logikai feldolgozó modul kimeneti adatait, ami nem más, mint maga a függőségi terv. Hogy az így létrejövő rajz kizárólag esztétikai értelemben milyen színvonalú, az stratégiai kérdés is. Egy jóváhagyó számára benyújtható tervnek ugyanis komoly esztétikai hagyományai vannak. Egy ilyen grafikus program megírása pedig óriási feladat. Ezt a hatalmas feladatot nem elvetve meg kell tehát teremteni a kapcsolatot a kereskedelmi CAD programok felé. Ez az első pillanatra ijesztőnek tűnő feladat azonban más szempontból kíváncsú is. A mai gyakorlatunk szerint ugyanis egy adat, pl. szelvéyszám, több rajzon is szerepel. A grafikus rajzok gépi összevetése, azaz fájlok analízisa tehát a tervezés minőségét javító feladat.

Az általunk megírt grafikus editor biztosítóberendezési értelemben vett objektum alapú. (Programozás technikai értelemben igyekszünk kerülni az objektumokat) Ez azt jelenti, hogy egyetlen beillesztéssel elhelyezhetünk a rajzfelületen egy, a függőségi terven használt szimbólumot. Ennek a szimbólumnak aztán a szimbolizált biztosítóberendezési objektumra jellemző tulajdonságai vannak. Ezek a tulajdonságok egyrészt az állomás vágányhálózataiból származóan, topológiai jellegűek, másrészt olyan pro-



1. ábra: Tervezési folyamatábra

jektálandó adatok, amelyek származhatnak forgalmi igényből, helyi sajátosságból, vagy az objektum helyzetéből nem következtethető függőségből. Ezen adatok helyességéért a projektáló (validáló) a felelős.

Az állomás matematikai modellezése gráffal, leírása értelemszerűen mátrixszal történik. A mátrix sorai, oszlopai az elemek kapcsolódási logikája szerint mutatnak egymásra.

Az editornak tehát (koordináták alapján) fel kell ismernie az elemek kapcsolati rendszerét, és azt a logikai modulnak rendelkezésre kell bocsátania.

A logikai modell alapja egy ún. „gráf motor” (működési elve hasonló az útvonaltervező programokéhoz). A „gráf mutató” a feladattól függő startpontból kiindulva be tudja járni a teljes gráfot (a vágányhálózatot), a bejárt vágányúti elemek (nódok) a feladat függvényében irányítják a keresést, illetve tesznek megjegyzéseket az útvonal leíró táblába.

Egy érdekes probléma a keresési módszer irányítása. Egy nagyobb állomáson, vagy egy hosszabb centralizált vonalszakaszon ugyan is egy összetett vágányút keresése esetén akár több tíz csúccsal álló váltót is kezelni kell. Az így adódó (pl. 2³⁰) lehetőség még a leggyorsabb számítógéppel sem kezelhető. A keresési korlátozásokra két megoldást dolgoztunk ki. Az egyik a célpont adatbázisában elhelyezett, a hozzávezető útvonalak projektálással történő korlátozása. Mivel ez manuális feladat, munkaigényes, magában hordozza a biztonságot nem veszélyeztető tévesztés lehetőségét. A második megoldás, a jó öreg váltóállító lánc modellezése. Az előfeldolgozást jelentő „váltóállító lánc” rutin lefuttatásával már „csak” az értelmes útvonalak között kell válogatni. Az alap vágányutak kiválasztására több lehetőséget próbáltunk kidolgozni, eredménytelenül. A számba jöhető elvek, mint pl. jobbra tarts, balra tarts, le a fővonalról, legkevesebb váltó a vágányútban, stb. algoritmizálása csak triviális esetben adtak volna elfogadható eredményt.

Mind a logikai modul feltételrendszerének kidolgozásakor, mind az eredmények ábrázolásakor a D70-es berendezés elveiből indultunk ki. A „gráf motor” megfelel a nyomkébeleknél, illetve az egyes egység típusoknak, míg a projektálandó feltételek az egységeken található program átkötéseknek. Természetesen nem arról van szó, hogy bármiféle kísérletet tettünk volna az egységek funkcióinak leprogramozására, de a D70-es egységek funkcióinak, de legfőképpen a programozási lehetőségeinek az elemzése sokat segített a lehetőségek és szükségességek pontos felmérésében.

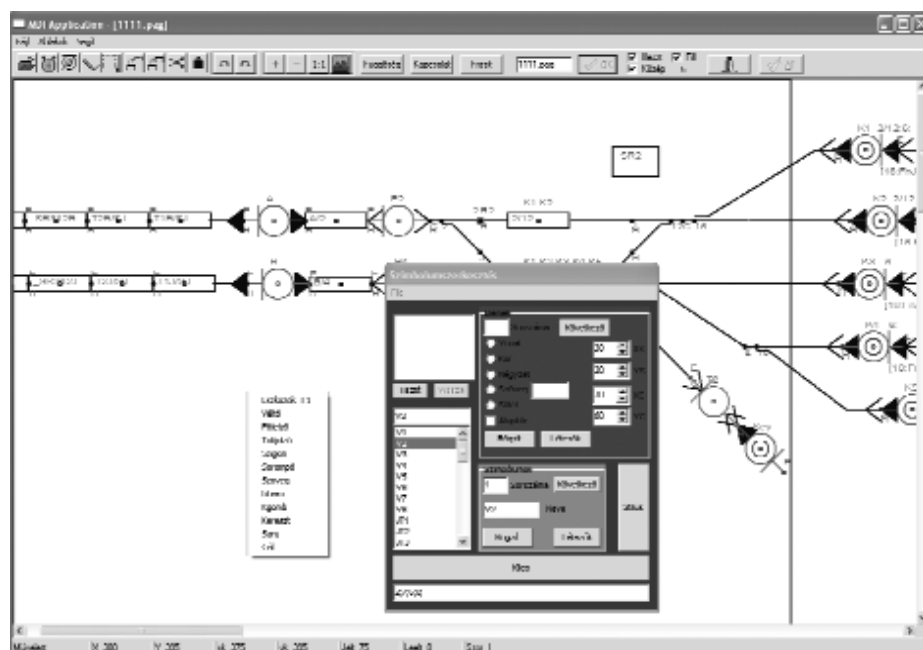
Figyelemmel kellett lenni azonban arra a nem elhanyagolható különbségre, hogy a MÁV-nál eddig alkalmazott elektronikus biztosítóberendezések menettervi kizárásainak megvalósítása filozófiájában tér el a D70-es elvtől. A jelfogós berendezéseink a veszélyes menetek kizárására oldalvédelem biztosítását, illetve célkizárást alkalmaznak. Leegyszerűsítve ez azt jelenti, hogy egy menet akkor állítható be, ha a célnál nincs menetkizárás, és ha az oldalvédelmi sávban lévő jelzők oldalvédelmet biztosítanak pl. célkizárással.

Az elektronikus biztosítóberendezések, bár ismerik a jelzővel adott oldalvédelem, vagy célkizárás fogalmát, az elengedő menetek kizárására megcsúszási vágányutakat alkalmaznak. Ez azt jelenti, hogy a céljelző mögött fekvő váltókat, vágányszakaszokat a megcsúszási vágányút végéig lezárják, vagy igénybe veszik. Az igénybe vétel (tükörfordítás németből) azt jelenti, hogy a váltó állítható marad, de rajta átvezető menet nem lehetséges. Az megcsúszási vágányúttal, megcsúszási zónával (így nevezzük az igénybe vett váltók összességét) kialakított függőségvizsgálat lényegesen egyszerűbb mechanizmusokat igényel, de bonyolítja a tervező tool funkcióit. Egy új feltétel rendszer szerint, generálni kell ugyanis a megcsúszási vágányutak/zónák adathalmazát. Az 50 m-es megcsúszási vágányutak generálása ugyanis meglepően sok feltételt tartalmaz. A rossz látási viszonyok esetére bevezetett 300 m-es megcsúszási vágányút keresése kevesebb feltétel vizsgálatot igényel, de mind a két esetben optimalizálási problémák lépnek fel. Az optimalizálás jelent-

heti a felesleges váltóállítás elkerülését, vagy a megcsúszási vágányút feloldásának két perces időzítéséből eredő forgalmi korlátozásokat elhagyását. Tipikus példa erre egy céljelzőtől induló több megcsúszási útvonal miatti menetkizárás elkerülhetősége. Ha egy megcsúszásban érintett váltó oldalvédelmet keres magának. Ezen oldalvédelmi váltót pedig ki lehet törölni a másik menet megcsúszási vágányútból. Optimalizáló rutinok specifikálása, írása, tesztelése pedig rendkívül körülményes feladat. Az automatikus optimalizálást ezért nem javasoljuk, de lehetőségének felajánlását igen. A vágányutak generálása bár a leglátványosabb feladat, számítástechnikai szempontból nem túl izgalmas. Az oldalvédelem vagy a vágányútdoldás kérdésköre a szerteágazó feltételrendszer miatt sokkal nagyobb feladat. Itt ugyanis nem elég a gráfban „futkározni”, hanem fel kell deríteni az egyes objektumok vágányhálózatban elfoglalt helyét, és az objektumok működését a helyzetüktől függően vezérelni.

FÜGGŐSÉGGENERÁLÓ MODUL

Egy rövidke cikk nem teszi lehetővé, hogy teljes mélységében foglalkozzak egy függőségi rendszer minden elemével. A lenn felsorolt és ismertetésre kerülő fontosabb függőségek csupán ízelítőül szolgálhatnak a gondolkodásmód megértéséhez. Igyekeztem összevetni a táblázatos és a grafikus ábrázolást, illetve a két elv filozófiáját. Összességében arra a megállapításra jutottam, hogy a táblázatok lényegesen finomabb leírást tesznek



2. ábra: Grafikus editor a szimbólumszerkesztő ablakkal

lehetővé, a grafikus ábrázolás megközelítése pedig sokkal szakmaibb, sokkal inkább támaszkodik a feldolgozó szakmai ismeretére. Ez az utóbbi, bár simogatja szakmai lelkünket, de gondoljunk arra, hogy a tervező monitorok előtt az esetek többségében programozó matematikusok ülnek. Ha szabad egy önfényező megjegyzést tennem: elégedettséggel tölt el, hogy sikerült eredményre vinni azt a gondolatot, „ha biztosítóberendezést építünk, a biztosítógondolkodás legyen a mértékadó, és ne a programozó matematikusé”. (A matematikusok persze továbbra is rendkívül fontos szerepet játszanak mind a projektálásban, mind a biztonságban).

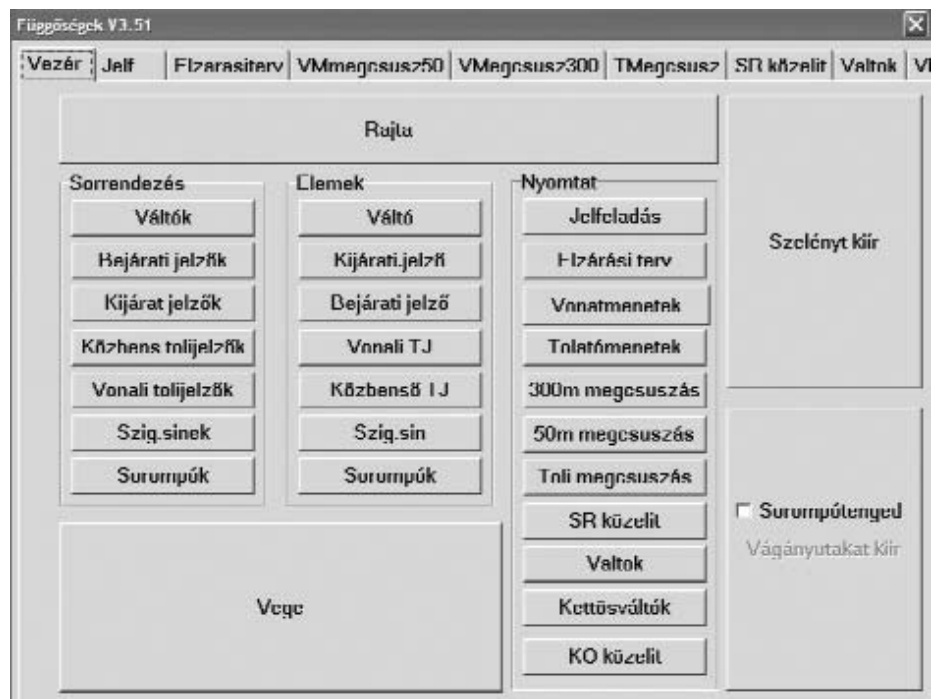
VÁLTÓELZÁRÁSI TERV

A SIMIS IS-nél alkalmazott lezárási táblázatok nagyban hasonlítanak a D55-nél megszokott táblázatokhoz. Lényeges eltérés, hogy a SIMIS IS-ben külön projektálható a különböző elemek lezárása és a foglaltságvizsgálat feltételei. Az elemek „igénybe vétele” új fogalom bevezetését is igényelte. A táblázatos elzárási terv érintett váltóinak grafikus ábrázolására nincs szükség, az a vágányképből egyértelműen kikövetkeztethető. Más a helyzet az oldalvédelemmel. A nyomvonal elves berendezéseknél nincs lehetőség az oldalvédelmi sávhatár helyének dinamikus meghatározására. Ha egy elemet oldalvédelemben ellenőrzött szakasz végének projektálunk, az minden oldalvédelmet kérő váltóra érvényes. Ez egy vágányúton belüli kettős terelő váltóknál jelenthet bizonytalanságot. Táblázatos függőségeírás esetén ez jól kezelhető, de szükségessé válhat a sávhatárt jelző pont mellett az oldalvédelmet kérő váltó nevének feltüntetése, vagy az oldalvédelmi sávhatár meghosszabbítására.

A lezárási táblázatok már utalást tartalmaznak a vágányút, illetve a cél oldási feltételére is.

A vágányutakkal kapcsolatos előtervi ábrázolási módok lényeges elvi problémát nem jelentenek, de az oldalvédelmi elemek ábrázolása – azaz hová kerüljenek a pontok, nyilas pontok – a képernyőn, a sok-sok lehetőség miatt egy szoftveres lelkének széklábfaragás.

A 4. ábra az elzárási táblázat(ok) egy kis részletét mutatja be. Táblázatos ábrázolásnál illetve feldolgozásnál nem célszerű grafikus szimbólumokat alkalmazni, ezért mindenütt karaktereket alkalmaztunk. Szembetűnő, hogy a feltételeket jelentő karakterek típusa lényegesen több, mint egy hagyományos elzárási terven. Már „belső” feltételeket is leírunk.



3. ábra: A függőségi modul funkcionális ablaka

MEGCSÚSZÁSI ELEMOK

A megcsúszási vágányutat/zónát alapvetően nem a vágányutakhoz kell rendelni, hanem a céljelzőhöz. A megcsúszási elemek keresésének algoritmizálása meglepően sok feltételt tartalmaz, és nem (rendkívül nagy munkával) optimalizálható. Ha a megcsúszási vágányutat csak az ellenséges menetek kizárására használjuk, a megcsúszási vágányútban fekvő váltó oldalvédelmet kereshet magának. A megcsúszási vágányút oldalvédelmi váltóját pedig ki lehet venni az ellenséges megcsúszási vágányút elemei közül, ha az nem zár ki pl. szemben álló tolatásjelzőnél egyidejű kettős megcsúszás miatt meneteket.

A megcsúszási vágányutak grafikus ábrázolására két módszert alakítottunk ki. Az egyik nyíllal jelöli a jelző nevével jelzett megcsúszási vágányút végét, míg a másik a céljelző mellé szöveges formában írva. Az előbbi 300 méteres, míg az utóbbi 50 méteres megcsúszás leírásánál célszerű alkalmazni.

VÁLTÓK, KISIKLASZTÓ SARUK, VÁGÁNYZÁRÓ SOROMPÓK

A váltók, mint topológiai elemek, nagy pontossággal fel tudják „ismerni” helyzetüket, működésük nem tartalmaz helyi sajátosságokat, ezért funkciójuk jól algoritmizálható. A ráfutási szakaszok, közbezárási vagy az oldási feltételek generálása tehát egyszerű feladat, ábrázolásuk sem jelent gondot. A kettős terelő váltók

leírása érdemel még említést, hisz funkcióban a vágányúthoz, generálásban pedig a váltóhoz tartozik. A kettős terelő váltók ezért önálló adatállományt (táblázatot) igénylenek, de grafikus ábrázolásuk hagyományos módon történhet.

JELZŐK

A jelzők, bár ugyan olyan topológiai elemek, mint a váltók, adatkezelés szempontjából sokkal szorosabban kapcsolódnak a vágányutakhoz. A jelző adatbázisa tartalmaz a vágányutak start-, célfeltételeit, időzítéseket. Ezeket az adatokat gyakran nem is lehet grafikus módon ábrázolni.

SOROMPÓK, KÖZELÍTÉSI VÁGÁNYUTAK

A sorompók előtervi adatai közül biztosítóberendezés szempontjából alapvetően két adat releváns. A fedező jelzők időzítése, illetve a behatási távolság meghatározása. Ez annyira általános feladat, hogy gyakorlatilag egy független sorompó adatokat számító programot integráltunk az editorba, illetve a függőséget generáló modulba. A sorompó közelítési vágányútjának generálásához, bekapcsolási pontjának meghatározásához alapvetően két módszer áll rendelkezésre. A „gráf motor” célpontja vagy a sorompószámításokból származtatható bekapcsolási szelvényszám elérése, vagy egy előre projektált bekapcsoló elem

Híjgázgáz V3.51																				
Vezér	Jell	Elzárasterv	VMmegcsúszt/50	VMmegcsúszt/300	TMmegcsúszt	SN küzelit	Valuk	VKüzelit	TKüzelit	Vuanmenetek										
.	.	.	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
.	.	.	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
A	01	5.8	J	J	R	J	J	J	R	-	J	-	-	-	-	-	-	-	-	-
A	04	5.8	J	J	R	J	J	J	R	-	B	-	-	-	-	-	-	-	-	-
A	05	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	06	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	07	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	08	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	09	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	10	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	11	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	12	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	13	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	14	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	15	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	16	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	17	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	18	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	19	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	20	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	21	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	22	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	23	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	24	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	25	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	26	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	27	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	28	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	29	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	30	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	31	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	32	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	33	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	34	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	35	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	36	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	37	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	38	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	39	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	40	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	41	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	42	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	43	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	44	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	45	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	46	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	47	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	48	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	49	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	50	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	51	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	52	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	53	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	54	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	55	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	56	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	57	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	58	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	59	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	60	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	61	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	62	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	63	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	64	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	65	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	66	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	67	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	68	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	69	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	70	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	71	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	72	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	73	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	74	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	75	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	76	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	77	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	78	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	79	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	80	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	81	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	82	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	83	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	84	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	85	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	86	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	87	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	88	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	89	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	90	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	91	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	92	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	93	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	94	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	95	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	96	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	97	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	98	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	99	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-
A	100	5.8	J	J	R	J	B	J	R	-	B	-	-	-	-	-	-	-	-	-

4. ábra: Váltóelzárás. Egy táblázat (elzárás) a sok közül

megtalálása. Táblázatos leírás esetén fel kell sorolni minden lehetséges közelítési vágányutat az azt alkotó elemekkel.

BIZTONSÁGI KÉRDÉSEK

A tervező eszközök biztonságáról, biztonsági szintjéről általában megoszlanak a vélemények. Abban egyetértés van, hogy alkalmazásuk valamilyen szinten biztonsági kérdés, de SIL besorolással nem rendelkeznek. Ennek egyik oka, hogy az általuk szolgáltatott adathalmaz konzisztenciáját a biztosítóberendezés ellenőrzi, másrészt olyan adat-halmaz, amely jelentős részét emberi tévesztés lehetőségével projektálják, nem feleltethető biztonsági osztályba sorolás követelményeinek. A tervezőeszközök tehát fontos szerepet