Eötvös Loránd Tudományegyetem
Informatikai Kar

Általános partíciós adatbázismodell. Funkcionális függőségi rendszerek elemzése

A doktori értekezés tézisei

Készítette: Molnár András

Témavezető: Dr. Benczúr András
Tanszékvezető, Információs Rendszerek Tanszék

Informatikai Doktori Iskola
Iskolavezető: Dr. Demetrovics János

Információs rendszerek doktori program
Programvezető: Dr. Benczúr András

2007.

1. Előzmények és célok

Egy adatbázis szerkezetének tervezése a valós világ modellezésén alapszik. Az *adatmodell* fogalma kétféle értelemben használatos: egyrészt jelölhet egy adatbázis-paradigmát, egy nyelvet absztrakt adattípusokkal, mellyel adatbázissémák írhatók le, ilyen pl. a hagyományos *relációs modell* [Cod70, Ull89, AHV95]. Egy adatbázis-kezelő rendszer absztrakciós szintet kínál az adatok kezeléséhez, az adatmodellhez igazodva, így a felhasználónak ill. alkalmazásfejlesztőnek nem kell törődnie pl. fizikai tárolással, vagy lekérdezési algoritmusokkal.

Az adatmodell másik értelmezése egy konkrét alkalmazási terület *fogalmi modellje*, mely a valós világ entitásait (objektumait), tulajdonságaikat és a köztük lévő kapcsolatokat írja le. Erre leggyakrabban az *egyed-kapcsolat modell*t használják [Ull89]. Egy kapcsolat típusát meghatározza a benne résztvevő entitások száma, a részvétel kötelező jellege egy adott egyedosztályra nézve, valamint az, hogy egy entitással hány másik entitás lehet kapcsolatban. Egy kapcsolat pontos specifikációjához, a későbbi redundanciák és anomáliák elkerüléséhez, a séma finomításához (normalizálásához) a valós világ elemzése során meg kell tudnunk határozni ezeket a számossági jellemzőket.

A számossági jellemzők egyik alapvető formája a *funkcionális függőség* [Cod70], amely az fejezi ki, hogy egy adattáblán belül egy vagy több attribútum (oszlop) értéke egyértelműen meghatároz valamely másik attribútumot. Bináris kapcsolatokra nézve ez azt jelenti, hogy az egyik osztály egy egyede legfeljebb egy egyeddel lehet kapcsolatban a másik osztályból. *Negált függőségként* jelenik meg, ha ezt nem írjuk elő (elsősorban teljesség- és konzisztenciaellenőrzés szempontjából). Ez alapján két egyedosztály között négyféle kapcsolat lehetséges, három alaptípusban (*1:1, m:n, 1:n* és vele egy típusba tartozó szimmetrikus párja az *n:1*).

Többágú kapcsolatok esetén az általánosan használt modellező nyelvek nem teszik lehetővé az összes lehetőség jelölését, sem a logikai következtetést, sem a konzisztenciaellenőrzést. A többágú kapcsolatokat a séma finomításakor gyakran dekomponálják, normalizálják. A legjobb dekompozíció érdekében, mindenekelőtt szükséges megtalálni az összes függőséget és így meghatározni a kapcsolat típusát. Pl. [Cam02] megmutatta, hogy a 3 ágú kapcsolatoknak 14 féle típusa lehetséges (melyek egy része dekomponálható). Kérdés, hogy mit mondhatunk nagyobb számok esetén [Tha87].

A funkcionális függőségek jól megalapozott, gazdag elméleti háttérrel rendelkeznek. Hagyományos jelölésrendszerük viszont olykor nehézkes, sok benne a redundancia. Éppen ezért munkám egyik célkitűzése az, hogy a függőségek kezelése egyszerűbbé, áttekinthetőbbé váljon, a függőségi rendszerek osztályozhatók legyenek, számuk meghatározható legyen. Ehhez alkalmas (pl. grafikus) reprezentációkat, következtetési rendszereket kell keresni.

Az utóbbi időben az *adattárházak, multidimenziós adatbázisok* [HK01] mind elméletben, mind gyakorlatban igen fontos szerephez jutottak. A *multidimenziós adatmodell* a valós világ eseményeit, megfigyeléseit (pl. tranzakciók, mérések) párhuzamosan különböző szempontok szerint osztályozza. Ezen szempontokat nevezzük *dimenzióknak*, s általában egy dimenzió hierarchikusan szervezett részletezettségi szintekből áll. Egy *adatkocka* olyan szerkezet, mely aggregált metrikus vagy leíró adatokat tartalmaz a dimenziók felbontási szintjeinek lehetséges értékeiből képzett kombinációkhoz.

A multidimenziós adatbázisok esetén általában feltételezzük, hogy a dimenzióhierarchia többékevésbé homogén és minden adat rendelkezésre áll a hierarchia legalsó szintjén, azaz a legfinomabb felbontás az egész adathalmazra ismert, metrikus adatokkal együtt. Az alkalmazás természetéből vagy információhiányból adódóan ez heterogén is lehet, pl. bizonyos dimenziók vagy felbontási szintek az adatoknak csak bizonyos részére értelmezhetők vagy ismertek. Ilyenkor fontos, hogy olyan sémát tudjunk definiálni, amely pontosan leírja a struktúrát és a rendelkezésre álló (*extenzionális*) adatokat, azokat minél jobban kihasználva a belőlük levezethető (*intenzionális*) adatokat, továbbá

hogy mik az értelmes, érvényesen elvégezhető navigációs és aggregációs műveletek, s biztosítja az adatbázis konzisztens bővítését arra az esetre, ha további adatok válnak ismertté.

Az aggregációk szempontjából fontos tulajdonság az *összegezhetőség*, azaz hogy egy halmazra aggregált érték előállítható-e a halmaz diszjunkt felbontásán ismert aggregált értékekből. Ehhez egyrészt annak biztos megállapítása szükséges, hogy az összegzendő értékek valóban diszjunkt halmazokra vontatkoznak. Másrészt, hogy az aggregáció szemantikája olyan-e, hogy az összegzés értelmes.

Az adatkocka felfogható úgy is, mint a valós világ egy populációjának diszjunkt halmazrendszerekbe való sorolása és az elemi halmazmetszetekhez aggregált adatok hozzárendelése. Ez tehát egy többszempontú, többszintű *partícionálás*. Spyratos [Spy87] a hagyományos relációs modellhez bevezetett egy speciális, *partíciós szemantikát*. Munkám másik alapvető célkitűzése ennek továbbfejlesztése, egy általános partíciós adatbázismodell lehetőségeinek kutatása, diszjunkt halmazrendszerek elnevezési struktúrájának és – erre épülve – aggregációs adatainak helyes kezelésére. A partíciók fogalma nem csak az adattárházak témakörében jelenik meg. A földrajzi információs rendszerekben a *geográfiai mezők* is többnyire partíciókként értelmezhetők.

A két célkitűzés szorosan kapcsolódik egymáshoz, mivel a partíciók finomítási struktúráját funkcionális függőségi rendszerekkel lehet általánosan megfogalmazni, a hagyományos dimenzióhierarchia ebbe speciális esetként illeszkedik.

2. Alkalmazott módszerek

Az általános partíciós adatbázismodell alapjainak lefektetéséhez elsősorban Spyratos logikai - dedukciós elvű partíciós modelljéből [Spy87] indultam ki, valamint az adatkocka fogalmából. A cél egy alkalmazásfüggetlen partíciós modell; speciálisan az adatkocka általánosítása heterogén struktúrákra, hiányos információra, később akár metaadatok tárolására.

A hagyományos relációs modellben a különböző attribútumértékek természetes módon a reláció sorait partícionálják. A partíciós szemantikával viszont egy további absztrakciós szint jelenik meg az adatmodellben, mivel a relációk nem a valós világ elemeiről közvetlenül, hanem azok halmazairól (elemi halmazmetszetekről) tárolnak adatot. A relációs adatmodell fogalmait újra kellett értelmeznem ezzel a szemantikával, feltárni a különbségeket a hagyományos esetekhez képest. Egy partíciórelációra vonatkozó fogalmat párhuzamosan, a partícionált elemekre (a populációra) nézve is értelmezni kellett. Annak lehetőségét vizsgáltam, hogyan lehet adatbázissá szervezni és műveleti készletet megadni partíciós szemantikájú relációkhoz. Ehhez jónéhány, a gyakorlatban előforduló alapesetet is felírtam és elemeztem.

A hagyományos relációs algebra műveleteit szisztematikusan megvizsgáltam, hogy a megszokott szintaxis milyen szemantikával értelmezhető, milyen externális ismeretek befolyásolhatják a a műveletek megvalósíthatóságát, s milyen új fogalmak bevezetésére van ehhez szükség.

A funkcionális függőségi rendszerek és a partíciók kapcsolatának vizsgálata során felhasználtam a Spyratos-féle konjunktív függőség fogalmát [Spy87], valamint a függőségi rendszerek ismert, de a gyakorlatban kevéssé alkalmazott félhálós (semilattice) reprezentációját [DLM89]. A félháló szerinti gráfreprezentációt használtam fel a partíciók struktúrájának leírására. Mindez motiválja a függőségi halmazok mélyebb elemzését.

A funkcionális függőségek hagyományos formalizmusában mutatok példát olyan kedvezőtlen esetre, amikor a jól ismert *Armstrong-axiómarendszerrel* [AHV95, Ull89] csak úgy lehet eljutni egy nemtriviális logikai következményhez, ha közben („majdnem") triviális függőségeket vezetünk le (a jobb oldal attribútumai között szerepel a bal oldal egy attribútuma).

A függőségi rendszerek számba vételéhez és áttekinthetőbb kezeléséhez bizonyos redundanciáktól megszabadulva, egyszerűsített szintaxist kellett definiálnom. Ehhez a hagyományos Armstrong-axiomatizációnál alkalmasabb szabályrendszerek keresése vált szükségessé, melyek helyességét és teljességét az egyszerűsített szintaxis keretein belül be kellett látni, felhasználva az Armstrong-axiómarendszer e tulajdonságait a hagyományos szintaxissal.

A módszer többek között lehetőséget ad adott attribútumszámra az összes lehetséges függőségi rendszer szisztematikus generálására, programmal.

Az áttekinthetőség érdekében a vizuális megjelenítés és logikai következtetés lehetőségeit is vizsgáltam. A függőségei rendszerek grafikus és táblázatos reprezentációját vezettem be és próbáltam általánosítani magasabb attribútumszámra.

3. Eredmények

3.1. Általános partíciós adatbázismodell

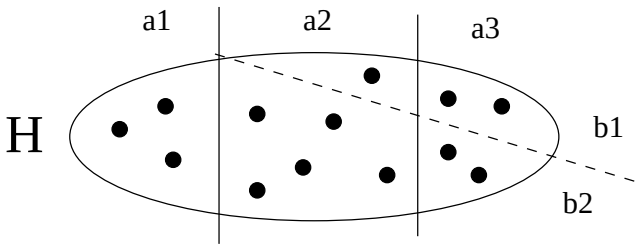
1. A disszertáció 4. fejezetében a Spyratos féle logikai-dedukciós *partíciós adatmodellt* [Spy87] fogalmazom át algebrai jellegű szintaxissal, s bővítem az *alaphalmazok, aggregációs attribútumok* fogalmával [1]. Ezzel a speciális szemantikával a jól ismert generikus relációs adatmodellből egy halmazleírás-kezelő nyelv válik, mely alkalmas a valóság véges, diszjunkt halmazrendszerekbe való sorolásának kódolására, a halmazelnevezések (kódok) kezelésére, statisztikai (aggregációs) adatok korrekt nyilvántartására (összegzésére), a diszjunktság alapelve szerint. Az aggregációk közül egyelőre csak a legegyszerűbbel, az összegezhető (*kumulálható*) aggregációval foglalkozom.

   Egy relációban előforduló elemi szimbólumok halmaznevek, az oszlopok partícióknak, a sorok halmazmetszeteknek felelnek meg. A partíciós adatbázis ezekből az elemi halmazmetszetekből építkezik, s a modell azt igyekszik jól jelölhetővé tenni, hogy melyek a diszjunkt halmazok. Az 1. ábra mutat példát egy kétattribútumos partíciórelációra, mely egy két szempontú partícionálást ír le a H alaphalmazon: az A szerinti partíció halmazai a_1, a_2, a_3, B szerint b_1, b_2, s ezek nemüres metszeteit tárolja az R reláció az N kumulatív attribútummal, mely most az elemszámra vonatkozik.

   A *partíciós szemantikát*, azaz a halmaznevek feloldását a szigorú értelemben vett partíciós adatbázis nem tartalmazza. Így azt tekintjük elvégezhető műveletnek, amit ezen a szinten meg tudunk oldani, a nyers (populáció szintű) adatokkal nem számolva, azokat külső ismeretként tekintve.

   Ennek eredményeképpen egy relációs leírási modellt kapunk, melyben a halmazok megnevezésében és diszjunktságának vizsgálatában kihasználható az adatok táblázatos megjelenése. Különböző relációk különböző alaphalmazok partícióit írhatják le és kumulálható aggregációs attribútumok tartozhatnak a relációkhoz. Így az egyszerű partíciós adatbázis fogalmához bevezetem a szemantika kompatibilitásának fogalmát, mellyel több reláció összemérhetőségét lehet rögzíteni, mivel pusztán a relációk alakjából nem következik, hogy a bennük szereplő attribútumok, szimbólumok (halmaznevek) milyen viszonyban vannak egymással.

2. Szintén a 4. fejezetben megmutatom, hogy a hagyományos relációs algebra naiv alkalmazása a partíciós szemantikával könnyen félrevezető lehet és a műveletek nem mindig adnak értelmezhető eredményt. Ennek elsődleges oka, hogy a relációkon felírt műveleteket valójában



R:

A	B	N
a1	b2	3
a2	b1	1
a2	b2	5
a3	b1	2
a3	b2	2

1. ábra. Példa partíciórelációra

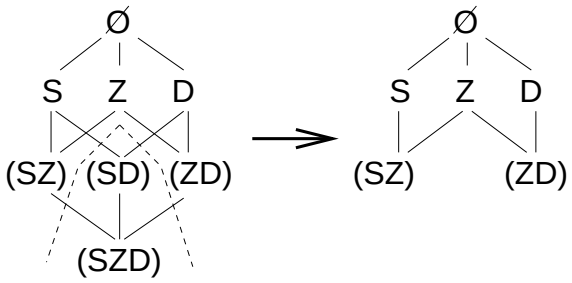
az általuk reprezentált halmazrendszereken akarjuk érteni. A *szelekció-projekció probléma* arra világít rá, hogy a partíciórelációk önmagukban nem mindig tartalmaznak elegendő információt egyes műveletek elvégzéséhez; külső ismeretre van szükség, hogy milyen viszonyú partíciókat írnak le. A *join probléma* pedig arra, hogy bizonyos esetekben nem határozható meg a teljes partícióháló egy adott halmazra, hanem összemérhetetlen partíciók lehetnek jelen, párhuzamosan.

A disszertációban felépítek egy partíciós algebrát, mely zárt a partíciórelációkra nézve, s a kumulatív attribútumokat a háttérben automatikusan kezeli. Ha ismert az operandusok partíciós szemantikája, meghatározható az eredmény szemantikája. A relációk alaphalmazait, ill. egymáshoz való viszonyukat nyilván kell tartani és figyelembe venni a kifejezések építésénél. Erre explicit lehetőségként bevezetem az *alaphalmaz-függőségeket*, melyek megszorítják a relációk partíciós szemantikáját, azaz hogy milyen halmazokat reprezentálhatnak a bennük előforduló szimbólumok. Az alaphalmaz-függőségek a szemantika kompatibilitásának szigorításaiként is tekinthetők. Mivel a konkrét hozzárendelést ezen a szinten nem tároljuk, egy alaphalmaz-függőség metaadatot jelent, ugyanakkor a relációkra nézve van szintaktikai vonzata is. A szintaktikai vonzat szükséges és elégséges feltételt jelent ahhoz, hogy adott relációkhoz létezzen az alaphalmaz-függőséget kielégítő partíciós szemantika. A különböző függőségi típusokhoz megadom azok szintaktikai vonzatait is [1].

3. A 4. fejezetben kitérek arra is, hogy funkcionális függőségekkel a partíciók finomítási struktúrája leírható és a függőségek az aggregálási, navigációs lehetőségeket is meghatározzák. Ezt részletesen az 5. fejezetben fejtem ki (ld. még [1]).

   A különböző elemi halmazokat leíró partíciók (partíciókombinációk, metszetpartíciók) éppen a függőségek szerinti zárt attribútumhalmazoknak felelnek meg. A zárt halmazok félhálója gráfként ábrázolható, mely éppen a partícióhálót reprezentálja, útjai a lehetséges aggregációs utak. Így ezt vezetem be a partíciós modellbe a partíciók szerkezeti gráfjaként.

   Az is leolvasható a szerkezeti gráfról, hogy egy attribútum elhagyása vagy hozzáadása mikor jelent valódi aggregációt vagy finomítást és mikor csak formai, elnevezésbeli különbséget (utóbbi eset akkor fordul elő, ha a kapott partíciós attribútumhalmaz lezártja megegyezik az eredetivel – ilyenkor biztosan nem történik valódi aggregációs művelet). Speciális esetben a szerkezeti gráf a szokásos részkockahálónak felel meg (mint pl. a 2. ábra bal oldalán a szaggatott vonaltól eltekintve, az S, Z, D partíciós attribútumok függőség nélküli partícióhálóját látjuk), dekomponálva pedig a dimenzióhierarchiáknak, de bonyolultabb struktúrák is leírhatók vele.



2. ábra. Teljes és parciális partíciós szerkezeti gráf

A fentieken túlmenően a partíciós szerkezeti gráf jelölésrendszerét általánosítom arra az esetre, amikor egyazon halmazra több, összemérhetetlen partíció ismert (*parciális gráf*): ld. a 2. ábrát, ahol a bal oldali teljes gráfot kapjuk, ha ismert az SZD szerinti partíció. Ha viszont csak az SZ, ZD szerintiek ismertek, akkor a szaggatott vonal alatti rész nem állítható elő, így kapjuk a jobboldali gráfot. A gráfjelölést tovább bővítem annak leírására, amikor egyes partíciók csak bizonyos részhalmazokra értelmezettek (*részhalmaz élek*). Ez indukálja az *ekvivalencia-élek* bevezetését a gráfban, s így akár lokális függőségek is deklarálhatók, bár e lehetőség még nincs kiaknázva teljes mértékben. .

4. Az 5. fejezet végén egy félautomatikus dekompozíciós módszert vázolok fel a partíciórelációs adatbázisséma meghatározására, adott szerkezeti gráf esetén. A tervezői interaktivitás egyes lépéseknél fontos lehet, mivel van lehetőség pl. arra, hogy hatékonysági okokból redundáns (denormalizált) dekompozíciót készítsünk.

   Ennek eredményeképpen a gráf csúcsaihoz partíciós algebrai kifejezéseket rendelhetünk. Mivel a gráf csúcsai a különböző lehetséges partícionálásokat fejezik ki, valójában nézetdefiníciókról van szó. A felhasználói navigációk (a hagyományos roll-up, drill-down, slice-hoz hasonló műveletek [HK01]) a gráfon értelmezhetők, alaphalmaz-választást (szelekció) és attribútumválasztást (projekció) jelölnek, melyek a gráf csúcsaihoz rendelt kifejezések segítségével automatikusan partíciós algebrába transzformálódnak. A felhasználói navigáció független a relációs reprezentációtól, hiszen a gráf csúcsaihoz rendelt nézetdefiníciós kifejezések lecserélhetők.

5. Az eddigiekre építve, a 6. fejezetben felvázolom egy általános, alkalmazásfüggetlen, négyszintű partíciós adatmodell javasolt formális kereteit. A négy szint: *1. strukturális (fogalmi), 2. logikai (adminisztrációs), 3. előfordulás (adat), 4. interpretációs.* A szintek kb. úgy értendők, mint a hagyományos relációs modellben a séma és az előfordulás. Ezek itt rendre a 2. és 3. szintnek felelnek meg. Az 1., strukturális szinten az értelmezhető partíciós struktúrát fogalmazhatjuk meg (szerkezeti gráfként), a 2. szinten azt, hogy ebből mi ismert, partíciórelációs adatbázis formájában, algebrai kifejezéseket rendelve a szerkezeti gráfhoz. Ezen a szinten tetszőleges (akár redundáns) dekompozíció lehetséges, nem befolyásolva az 1. szinten megfogalmazott felhasználói műveleteket. A 3. szint a tényleges adatok (partícióreláció) szintje, a 4. szint a partíciós szemantikáé, amely a relációkban szereplő szimbólumok mint halmaznevek feloldását jelenti. A négy szintű formalizmus egyes strukturális elemei elvben lecserélhetők, bővíthetők (*plug-in*): a halmaznevek szimbólumkészletének algebrája, a halmazdefiníciós és partíciódefiníciós kifejezések formalizmusa, valamint az aggregációdefiníció lehetősége.

6. A partíciós modell lehetőségeinek vizsgálatához a disszertáció 6. fejezetében számos egyszerű alapesetet írok fel, melyek mutatják a partíciós modell leíró lehetőségeit (pl. egy adatkockából tetszőleges rész kiemelése és tovább-bontása új szempont szerint). A konkrét alapesetek elemzése nyomán nyomán elemi építkező és lekérdező műveleteket fogalmazok meg a partíciós adatbázismodellben (*partíciós alapnyelv*), procedurális stílusban (ld. *A* függelék). A szintaxis szignatúrákkal adott, a szemantika egyelőre informálisan. A műveleti készletben jól meghatározható, hogy minek melyik szintre van hatása. Az első szintű műveletek az attribútumok és alaphalmazok alapján működnek, függetlenül a konkrét relációs reprezentációtól. A definiált műveletek az alapesetek leírásához elegendőek.

   Az így kialakult absztrakt nyelv és műveleti készlet tekinthető az általános partíciós adatbázismodell alapjának. Alkalmas arra, hogy egyszerű multidimenzionális sémákat megfogalmazunk benne, valamint hogy heterogén szerkezetű partíciórendszereket, hiányos információt kezeljen. A fejezetben kitérek a jövőbeni pontosítási, továbbfejlesztési lehetőségekre is, mint pl. a formális alapok teljes kidolgozása, a szintek egymásra hatásának és a lokális függőségek hatásainak elemzése, az adatok módosításának kérdése, elosztott partíciós adatbázisok lehetősége, a plug-inek pontos specifikációja, kifinomult logikai következtetési rendszer a 2. szintre, műveletek teljesebbé tétele, aggregációs lehetőségek általánosítása, s a modell alkalmazhatósága konkrét, valós méretű példákon, ehhez az igények meghatározása. A partíciós modell hosszabb távon akár az adattárházak és az aggregációk elméletét formálisan leíró paradigmává is válhat.

3.2. Funkcionális függőségi rendszerek vizsgálata

1. Az egyed-kapcsolat modellben [Ull89] és a hasonló grafikus fogalmi modellező nyelvekben a többágú kapcsolatok megadásának lehetősége korlátozott, a pontos specifikációhoz és a séma további finomításához, normalizáláshoz szükség van további adatbázis-függőségek reprezentációjára. Cél, hogy a függőségeket ne egyenként, hanem áttekinthető rendszerben, halmazként reprezentáljuk.

   A funkcionális függőségek hagyományos formalizmusa – különösen az Armstrong-féle axiómarendszerrel [AV85] ill. annak negált függőségekre való kiterjesztett változatával [Tha00] történő levezetés – gyakran nehézkes, redundáns. Ezen javítottam úgy, hogy a triviális és a jobb oldalon több attribútumot tartalmazó függőségeket szintaktikailag kizártam. Az Armstrong-axiómarendszer naiv szűkítése erre a szintaxisra viszont már nem teljes.

   A disszertáció 7. fejezetében az egyszerűsített szintaxishoz többek között az alábbi axiomatizációt adom meg, s igazolom helyességét és teljességét (*ST ill. PQRST szabályrendszer*, ld. még [2]). Y az egyetlen halmazváltozó, míg A, B, C páronként különböző attribútumokat jelölnek, melyek nem elemei Y-nak.

   (S) $\frac{Y \to B}{YC \to B}$ (T) $\frac{Y \to A, YA \to B}{Y \to B}$

   (P) $\frac{YC \nrightarrow B}{Y \nrightarrow B}$ (Q) $\frac{Y \to A, Y \nrightarrow B}{YA \nrightarrow B}$

   (R) $\frac{YA \to B, Y \nrightarrow B}{Y \nrightarrow A}$ ($\square$) $\neg(Y \to B, Y \nrightarrow B)$

   A helyességet és a teljességet a kiterjesztett Armstrong-axiomatizációra való visszavezetéssel igazolom, több lépésben (a szintén általam bevezetett *U szabályrendszeren*, ill. negáltakra

az *UE, NST* rendszereken keresztül). Az ST teljességéhez pl. konstrukcióval megmutatom, hogy az Armstrong-szabályrendszerben felírt tetszőleges levezetéshez létezik az egyszerűsített szintaxis fölött értelmezett U szabályrendszerben felírt, ekvivalens levezetés, s az U rendszer ekvivalens ST-vel.

A szabályrendszer több előnyös tulajdonsága között figyelemre méltó, hogy létezik a szabályoknak egy meghatározott sorrendje (reguláris kifejezéssel $(S)^*;(T)^*;(R)^*;((P)\|(Q))^*$), ami teljes levezetési módszert ad (ST/STRPQ algoritmus). Ezt is igazolom a 7. fejezetben (3. tétel), konstrukcióval. Szintén a szabályok alapján adok módszert (10. fejezet), mely egy függőség hozzáadását végzi el zárt függőségi halmazra [4].

Az osztályozás szempontjából bevezetem a függőség *dimenziójának* fogalmát (a bal oldal attribútumszáma), valamint az attribútum dimenziójának fogalmát (az őt funkcionálisan meghatározó minimális attribútumhalmaz elemszáma - ha ilyen nincs, akkor ∞).

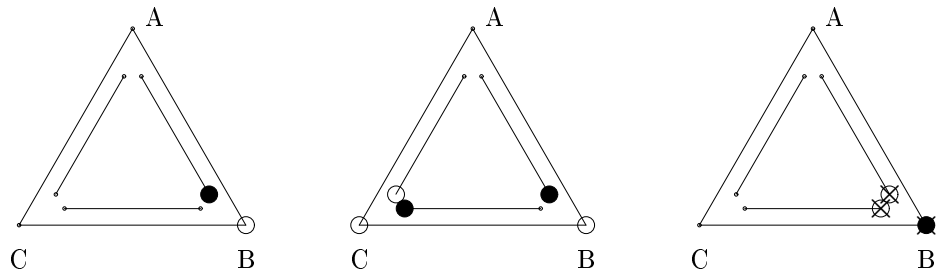
2. Kis attribútumszámra bevezettem a függőségi halmazok ábrázolására táblázatos ill. grafikus (*háromszöges*) reprezentációt (8. ill. 9. fejezet, ld. még [2, 3, 4, 6, 5]). Minden függőségnek a táblázat egy-egy cellája ill. geometriai alakzatok egy-egy csúcspontja felel meg.

   A 3. ábra mutat néhány példát a háromszöges reprezentációra. A bal oldalon $A \rightarrow B$ látható fekete pontként és implikációja, $AC \rightarrow B$ üres körrel jelölve; míg a középső ábrán az $\{A \rightarrow B,\ B \rightarrow C\}$ által generált függőségi rendszer. A jobb oldalon a $AC \nrightarrow B$ negált függőség és implikációi, $A \nrightarrow B$ és $C \nrightarrow B$.

   A reprezentáció elvben általánosítható nagyobb attribútumszámra is: n attribútum háromszöges reprezentációja az $n-1$ dimenziós térben létezik. Így pl. $n = 4$-re tetraédert kapunk, mely síkba transzformálható. Az átalakítás a tér dimenziójának növekedésével egyre nehezebb. A magasabbfokú reprezentációk elméleti jelentőségük mellett kiindulópontként szolgálhatnak egy függőségi rendszerek hatékony tárolására szolgáló adatszerkezethez.

   A levezetési szabályok adaptálásával logikai következtetést is bemutatok ezekben a reprezentációkban (pl. *grafikus következtetés*). A zárt attribútumhalmazok félhálójára [DLM89] vonatkozóan (mely egy alternatív grafikus reprezentációként tekinthető és a partíciós modellben a szerkezeti gráf szerepét tölti be) oda-vissza konverziós módszereket adok meg (10. fejezet).

   Mindez lehetővé teszi a függőségi rendszerek további vizsgálatát és csoportosítását, egy a tervezést segítő jövőbeni szoftver eszköz kifejlesztését. Elképzelhető, hogy hasonló megközelítéssel kezelhetők másfajta (pl. többértékű) függőségi rendszerek is.



3. ábra. Példák függőségi rendszerek háromszöges reprezentációjára

n	$\lvert\mathcal{SD}_n\rvert$	$\lvert\mathcal{SD}_n/\tau\rvert$	$\lvert\mathcal{SD}^0_n\rvert$	$\lvert\mathcal{SD}^0_n/\tau\rvert$
1	1	*1*	2	2
2	4	*3*	7	5
3	45	*14*	61	19
4	2 271	*165*	2 480	184
5	1 373 701	*14 480*	1 385 552	14 664

1. táblázat. A lehetséges zárt funkcionális függőségi halmazok száma n attribútumra. A dőlt betűs oszlop mutatja a konstans attribútumdefiníciót nem tartalmazó osztályok számát, melyek az n ágú kapcsolatoknak felel meg

3. A függőségi rendszerek számának meghatározásához készítettem egy PROLOG programot, melyet a disszertációhoz mellékelek [2]. Az eljárás az ST szabályrendszer és az attribútumok dimenziói alapján szisztematikusan generálja a lehetséges függőségi rendszereket. Lényege, hogy mindig minimális bal oldalú függőségeket generál, s csak az (S) bővítési szabályt alkalmazza. A függőségi halmazt elvetjük, ha a (T) redukciós szabály alkalmazható, mivel az ilyen halmazok generálódnak úgy is, ha minimális baloldalakból kiindulva csak (S)-et alkalmazzuk. A programot optimalizáltam és így 3, 4, 5 attribútumra sikerült lefuttatni. Az eredményeket az 1. táblázat tartalmazza [3, 4].

   A táblázat első oszlopában a lehetséges zárt funkcionális függőségi halmazok száma látható, n attribútumra (konstans attribútumot definiáló függőségeket nem engedve meg). Ha az attribútumok szerepét nem tekintjük rögzítettnek, akkor megkapjuk a permutáció erejéig ekvivalens osztályokat, mint a függőségi rendszerek (többágú kapcsolatok) lehetséges típusait. A program e típusok számát is meghatározza, ezek a táblázat második oszlopában láthatók.

   Ha megengedünk konstans attribútumokat, azaz 0 dimenziós függőségeket, akkor a fenti elemszámok a táblázat harmadik és negyedik oszlopa szerint alakulnak.

   A kapcsolattípusok és függőségi rendszerek számára nagyobb n esetén egyelőre csak becslések ismertek [BDK+91, DLM89]. Időközben megjelent [HN05], amely a négy általam vizsgált érték közül egyet meghatároz $n = 6$-ra.

A szerző publikációi

[1] A. Benczúr and A. Molnár. An extended partition model for generalized multidimensional data. Technical Report 2/2007, eScience Regional Knowledge Center, Eötvös L. University, Budapest, 2007.

[2] J. Demetrovics, A. Molnár, and B. Thalheim. Graphical and spreadsheet reasoning for sets of functional dependencies. Technical Report 0404, Kiel University, Computer Science Institute, http://www.informatik.uni-kiel.de/reports/2004/0404.html, 2004.

[3] J. Demetrovics, A. Molnár, and B. Thalheim. Graphical reasoning for sets of functional dependencies. In *Proceedings of ER 2004, Lecture Notes in Computer Science 3288*, pages 166–179, Shanghai, China, 2004. Springer Verlag.

[4] J. Demetrovics, A. Molnár, and B. Thalheim. Relationship design using spreadsheet reasoning for sets of functional dependencies. In *Proceedings of ADBIS 2006, Lecture Notes in Computer Science 4152*, pages 108–123, Thessaloniki, Greece, 2006. Springer Verlag.

[5] J. Demetrovics, A. Molnár, and B. Thalheim. Relációs adatbázisok funkcionális függőségeinek grafikus axiomatizációja. *Alkalmazott matematikai lapok*, 2007, Budapest (accepted for publication).

[6] J. Demetrovics, A. Molnár, and B. Thalheim. Graphs representing sets of functional dependencies. *Annales Univ. Sci. Budapest, Sectio Computatorica*, 28, 2008, (accepted for publication).

További irodalom (kivonat)

[AHV95] S. Abiteboul, R. Hull, and V. Vianu. *Foundations of databases*. Addison-Wesley, Reading, MA, 1995.

[AV85] S. Abiteboul and V. Vianu. Transactions and integrity constraints. In *Proc. 4th ACM SIGACT-SIGMOD Symp. on Principles of Database Systems - PODS'85*, pages 193–204, Portland, Oregon, 1985. ACM Press, New York.

[BDK+91] G. Burosch, J. Demetrovics, G. O. H. Katona, D. J. Kleitman, and A. A. Sapozhenko. On the number of databases and closure operations. *TCS*, 78(2):377–381, 1991.

[Cam02] R. Camps. Transforming n-ary relationships to database schemas: An old and forgotten problem. Technical Report LSI-5-02R, Universitat Politècnica de Catalunya, 2002.

[Cod70] E. F. Codd. A relational model for large shared data banks. *CACM*, 13(6):197–204, 1970.

[DLM89] J. Demetrovics, L. O. Libkin, and I. B. Muchnik. Functional dependencies and the semilattice of closed classes. In *Proc. MFDBS'89, LNCS 364*, pages 136–147, 1989.

[HK01] J. Han and M. Kamber. *Data Mining: Concepts and Techniques*. Academic Press, Morgan Kaufmann Publishers, San Diego, USA, 2001.

[HN05] N. Habib and L. Nourine. The number of moore families on n=6. *Discrete Mathematics*, 294(3):291–296, 2005.

[Spy87] N. Spyratos. The partition model: A deductive data base model. *ACM Transactions on Database Systems*, 12(1):1–37, 1987.

[Tha87] B. Thalheim. Open problems in relational database theory. *Bull. EATCS*, 32:336 – 337, 1987.

[Tha00] B. Thalheim. *Entity-relationship modeling – Foundations of database technology*. Springer, Berlin, 2000. See also http://www.informatik.tu-cottbus.de/∼thalheim/HERM.htm.

[Ull89] J. D. Ullman. *Principles of database and knowledge-base systems*. Computer Science Press, Rockville, MD, 1989.