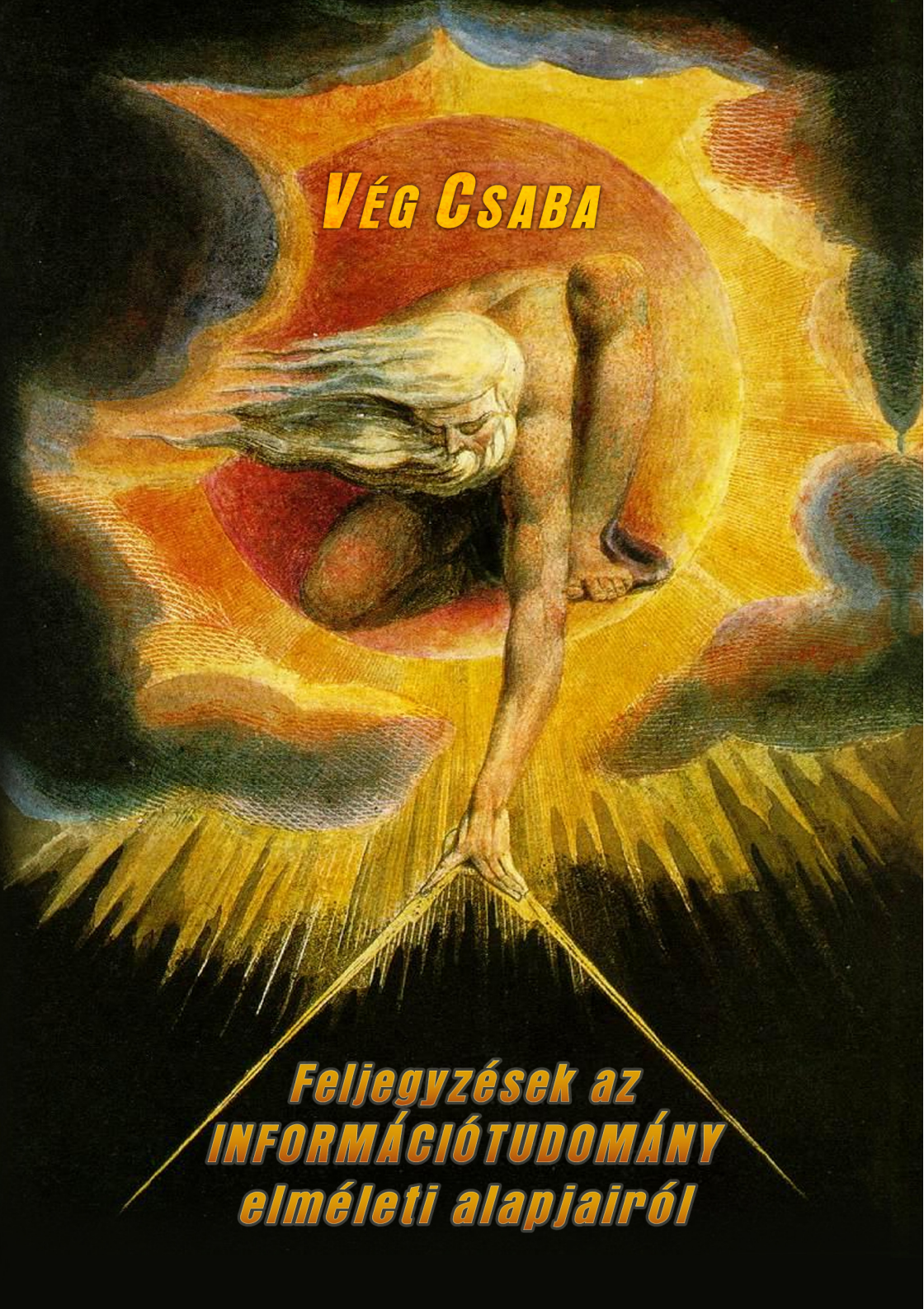




VÉG CSABA

***Feljegyzések az
INFORMÁCIÓTUDOMÁNY
elméleti alapjairól***

VÉG CSABA

**Feljegyzések az
INFORMÁCIÓTUDOMÁNY
elméleti alapjairól**

Budapest, 2008.

© Vég Csaba, 2008

Első kiadás, 2008. augusztus

Minden jog fenntartva. A szerző előzetes írásbeli engedélye nélkül jelen könyvet vagy annak részleteit más nyelvre lefordítani, valamint bármilyen formátumban vagy eszközzel reprodukálni, tárolni és közölni tilos.

Az UML az Object Management Group védjegye.

A könyvben védjegyek és bejegyzett védjegyek nagy kezdőbetűvel vannak szedve.

ISBN: 978-963-87743-2-3

Kiadó: Logos 2000 Bt.

Debrecen

<http://www.logos2000.hu>

Borítóterv: Csukás Csaba

(William Blake: The Ancient of Days (1794))

Ada Byron Lovelace
emlékének

Az informatikát, azon belül is az információs rendszerek fejlesztését tekinthetjük egy olyan katalizátornak, amely problémáival, illetve eszközkészletével segít kialakítani egy tiszta fogalomrendszert az információ ábrázolásáról, illetve az ismeretszerzésről. A számítógép így teremti meg azt az alkímiai laboratóriumot, amely a pszichénél összehasonlíthatatlanul egyszerűbb felépítésével, a reprezentáció megtekinthető formájával, a rendkívül primitív tároló és műveleti alapeszközökkel, a megfogalmazások (programok és adatok) nagy bonyolultságának, ugyanakkor tökéletes pontosságának az igényével lehetővé teszi, hogy az információ reprezentációival „vegytiszta” körülmények között szembesüljünk. Ezek a körülmények hasonlíthatók a telekommunikáció által megteremtett „vegytisztasághoz”, amely az átviteli, azaz a mennyiségi oldalról közlítő információelméletet megteremtette.

Célunk egy olyan elvi megközelítés felvázolása, amely egyszer elvezethet a rendszerfejlesztés egzakt fogalmi kereteinek kialakításához, valamint az általános értelemben vett ismeretszerzés és reprezentáció legalapvetőbb fogalmainak matematikai precizitású tisztázásához.

Elméleti megközelítésünk célja a fogalomrendszer kiépítése, a változás belső összefüggéseinek megállapítása, valamint az ábrázolás minőségi fokozatainak és alternatíváinak a meghatározása. A megismerés alapeszközei a rendszerből bizonyos jellegek leválasztása, majd az önálló vizsgálatok eredményeinek egyesítése. Az elemekből építkezés legfelső szintjén a strukturálistipizálás, alsó szinten a szimbolikus ábrázolás, köztes szinten pedig a hatásokon és hasonlóságokon, illetve általában a szabályokon alapuló meghatározottságként történő ábrázolást alkalmazhatjuk.

Vég Csaba (1992-2007)

Tartalom

Bevezetés	1
Informatika	1
Az információ tudományai	2
Objektumorientált szemlélet, OMT, UML	5
Elméleti megközelítés	7
Fogalmak és szemlélet	9
Szemlélet	9
Hipotézisek	13
Alkalmazott jelölések	15
<i>I</i> Absztrakció és kompozíció	17
Rendszer	17
Determinisztikus megadású rendszer	22
Ekvivalencia	23
Kompozíció és felbontás	26
Feltétel, következmény, függetlenség és ellentmondás	32
Korlátozó és példány	38
Absztrakció és pontosítás	40
Típus	43
Makrorendszerek	46
Korlátozás és kiterjesztés	47
<i>II</i> Információ és ismeretszerzés	49
Információ	49
Ábrázolás	52
Szempont, pontosság és eltérés	61
Ismeretszerzés	67
Példákon keresztül történő ismeretszerzés	69
Analízis és szintézis	75
Absztrakciós és pontosító kiegészítés	78
A lehetséges és tényleges rész	80
<i>III</i> Racionalitás és analógia	87
Modell	87
Kvantumrendszer és kvantált rendszer	90
Kvantumváltozás	92
Ismeretszerzés mint rendszer	94
Ismeretszerzés mint kiválasztás	97

Lépésenkénti véges választású közelítés	98
Állapot lépésenkénti véges közelítése	100
Szimbolikus modell	101
Racionalitás és analógia	106

***N* Hatás és kapcsolat 111**

Absztrakt megadású rendszer	111
Átírás nem absztrakt megadású rendszerré	116
Működések feltétele és korlátozója	118
Entitás korlátozója	120
Entitáskorlátozók erőssége	123
Hatás, függés, kapcsolat	124
Hatások erőssége	130
Enitás, környezete és külvilága	133

***V* Időben lezajló változás 137**

Objektumok rendszere	137
Esemény és működés	139
Transzformáció: átmenet és akció	141
Interakciók	144
Tevékenység	149
Absztrakt állapot, esemény és módosító	153
Vezérlés: működés algoritmikus környezetben	159

***VZ* Belső rendszer 163**

A belső rendszer	163
Struktúra	164
Tipizálás, fogalmi rendszer	166
Hatás ellensúlyozása, kohézió	169
Implicit és explicit kapcsolat	172
Együtműködés	176
Környezet	180
Strukturálási szintek	185
Szabályrendszer	188
Rendszer ábrázolása	193

***a* Objektumorientált szemlélet 199**

A valóság modellezése	199
Absztrakció és az OMT modelljei	200
Objektum: attribútumok és műveletek	203
Osztályok közötti viszonyok	207
Dinamikus modell	216
Funkcionális modell	222
Absztrakció és kompozíció	223

Ⅹ. Intelligencia, mint a megértés és az absztrakció képessége	227
Mesterséges intelligencia	227
Mit is értünk azon, hogy intelligencia?	228
Intelligencia	229

Formátumkonvenciók

- A *definíciókat* a ∇ jel vezeti be. A definíción belül *dőlt betű* emeli ki a definiált fogalmak elnevezéseit. A közbevetett megjegyzések és kiegészítő tulajdonságok zárójelek között szerepelnek.
- Az *állításokat* a § jel és vastag betűs cím vezeti be, mely cím az állítás összefoglaló elnevezése.
- A *megjegyzéseket*, a $\curvearrowright$ jel és dőlt betűs mondat jelöli.

Matematikai jelölések:

- Az „akkor és csak akkor, ha” állításokat a „*pontosan akkor, ha*” szókapcsolat írja le.
- Természetes számnak a 0 számot és rákövetkezőit tekintjük.
- D_f jelöli az $f : A \rightarrow B$ függvény értékkészletét, azaz B -t.
- Az $f : A \rightarrow B$ függvény, illetve $A' \subseteq A$ esetén az $f(A')$ a B azon értékeit jelöli melyekre $\{b \in B \mid \exists a \in A' : f(a) = b\}$. A függvény ekkor az A' kiindulási halmaz alapján „kiválasztja” a B értékkészlet $f(A') \subseteq B$ részét.
- Az $a_{i \in I}$ az egyszerűsített jelölése az indexelt a_i elemeknek, valamint egy I indexhalmaznak, ahol az i sorra felveszi az I értékeit. Műveletek, pl. $\bigcup_{i \in I}$ esetén a hagyományos értelmezést használjuk, tehát ebben a példában: „ i sorra felveszi I indexhalmaz elemeit”.

Bevezetés

A bevezetésben áttekintjük azt a folyamatot, amely a számítógépek és a számítástechnika megjelenésétől, a már nevében sem elsősorban az eszköz működését, hanem annak használatát és felhasználhatóságát előtérbe helyező informatikáig vezetett. Az informatika egyik fő problémájának az ismeretek megszerzését és reprezentációját tekintjük, így sorra vesszük az ezzel a területtel rokonítható tudományágakat. Ezután egy rövid összefoglaló következik a napjainkban a legelterjedtebben használt fejlesztési módszer, az objektumorientált szemlélet történetéről.

Informatika

A napjainkra már „múltnak” nevezhető XX. század derekán a technológiai eszközök színpadának rohamosan szaporodó szereplői között megjelentek az elektronikus számítógépek, amelyek már nem csak egyedi-kísérleti, hanem széleskörű ipari körülmények között is alkalmazhatóak voltak, sőt, az utolsó évtizedben már mindennapjaink szerves részévé váltak. A megjelenés a csúcspontja volt a száznál is több év elméleti kutatásainak és a korai kísérleti gépeknek, ugyanakkor megnyitotta egy teljesen új megközelítési mód kapuját, amelyben az elvi, illetve kísérleti szempontok mellett a gyakorlat és az alkalmazhatóság egyre nagyobb hangsúlyt kapott. A diszciplína összefoglaló jellegű „számítógép-technika” elnevezését hamarosan az információ-technológia, illetve az *informatika* elnevezés váltotta fel, amely jelzi, hogy a hangsúly a „gépről”, annak szerkezeti-technikai felépítéséről, sőt, a „számítási műveletekről”, áthelyeződött a számítógép ezeken túlmutató lehetőségeire, az információ tárolására és feldolgozására.

A robbanásszerű elterjedés forradalma több, nem-technikai problémát is felvetett. Az információ „vegytisztá” állapotban való megjelenése, annak másolható, többszörözhető volta, sőt, nyitottsága az elemezhetőség és az azonos értékűvé módosíthatóság (pl. program-feltörés) értelmében, mindennapjainkból jól ismert szerzői-jogi problémákat is eredményezett.

A „nagysebességű műveleti kapacitás” széleskörű elérhetősége magát az emberi pszichét is egy rendkívül érdekes és kényes probléma elé állította. A

számítógép alapvetően csak nagyon primitív számítási és tárolási/visszatöltési utasítások végrehajtására képes. A rendelkezésre álló műveleti sebesség és tárolási kapacitás azonban rendkívül összetett feladatok megadását is lehetővé tenné. A probléma tehát az, hogy egy átlagos körüli intelligenciával, amely egyszerre mindössze 5-7-9 alkotórészt és azok viszonyait képes áttekinteni, hogyan lehet primitív elemekből rendkívül összetett rendszereket tökéletes pontossággal felépíteni. Maga a kérdés a számítógép felhasználhatóságának emberi korlátaira is vonatkozik, hogy lehetséges-e ipari körülmények között, megjósolhatóan, és személyekhez kötött kvalitásoktól többé-kevésbé függetlenül, információs rendszereket fejleszteni. A probléma a hetvenes években szoftver-krízisként kulminálódott, amikor a megkezdett fejlesztéseknek több mint 70 százaléka (!) félbeszakadt. Ettől a sokkoló időszakról kezdve a számítógép rendkívüli lehetőségei az emberi pszichét folyamatosan szembesítik intelligenciájának és áttekintőképességének korlátozott voltával. Ahogy egy korai programszervezési technika, a strukturált programozás egyik szerzője, Dijkstra megfogalmazta: a fejlesztés „a bonyolultság megszervezésének művészetévé” vált.

Hogyan tudunk összetett és bonyolult összefüggésekkel rendelkező elemeket, azaz – a közismert elnevezéssel élve – egy *rendszert* összeállítani? Hogyan tudjuk a bonyolultságot megszervezni? A kérdés a rendszerek megadásának, más néven *reprezentációjának* a lehetőségeire irányítja figyelmünket, amely az elemek és azok összefüggéseinek a megadását jelenti. Megadáson, „specifikáláson” valójában a rendszerre vonatkozó *információk* megadását értjük. Ez a problémakör tehát a (rendszerre vonatkozó) információ reprezentációjának témájával írható le, amelybe beleértjük a reprezentációk ellenőrzésének, módosításának, összehasonlításának, stb. területeit is.

Az információ tudományai

Könnyen felismerhető, hogy az információ reprezentációjának témaköre jóval túlmutat a számítógépekkel kapcsolatos mindennapi problémákon. Az ismeretek reprezentációit és elrendezésének lehetőségeit a számítógépek világától függetlenül is értelmezhetjük, amely a filozófián belül az *episztemológia* (ismeretelmélet) területe.

Az ismeretek reprezentációjának problémái összefonódnak azok ellenőrzésével, illetve az ismereteknek az ellenőrzés eredményétől függő módosítá-

sával. Az egyre pontosabb ismeretek megszerzése valójában egy tanulási folyamatot ír le, amely így a *megismeréstudomány* (cognitive science) közelébe vezet el bennünket. A megismeréstudomány alapvetően a psziché megismerőtevékenységét kutatja, így elsősorban a pszichológia oldaláról közelít.

Az információ fogalma a tudományos nyelvezetben a Shannon és Weaver-féle *információelmélet* megjelenésével vált teljes jogú polgárrá. Az elmélet születésének időszakában a technika egyik központi problémája az adatátvitel, a telekommunikáció, illetve a kódolás volt. Pontosán ezen problémák megfogalmazásának igénye teremtette meg az információelméletet. Ezek a témák azonban elsősorban átviteli, illetve esetlegesen tárolási szempontból tekintették az információt, így ez az információ-fogalom kizárólagosan csak egy mennyiségi aspektust ír le. Az elmélet alapján egy adott esemény információ-mennyisége a bekövetkezésének valószínűségéből számítható, mégpedig annál több információt hordoz egy esemény, minél valószínűtlenebb volt korábban. Ebből a szempontból az információ a váratlanság (mennyiségi!) mérőszáma lesz, mivel az információelmélet szerint az esemény bekövetkezésének a valószínűsége az *egyetlen* dolog, amit egy eseményről mérni tudunk. Az informatikában a *mennyiségi* szempontok mellett egyre inkább az információ szervezettségének *minőségi* szempontjai kerültek előtérbe. Egy információs rendszerben egy esemény jelentőségét nem elsősorban a valószínűsége határozza meg (bár ez is egy szempont), hanem az, hogy milyen jellegű és mértékű változást eredményezhet a rendszerben.

Az összetett és összefüggő alkotóelemekből álló rendszer fogalma a *rendszereken alapuló szemlélethez* (systems thinking) vezet el. A XX. század negyvenes éveiben a biológus von Bertalanffy által eredetileg felvázolt elméleti megközelítés azóta több ágra bomlott szét. Az egyik ág, a *rendszerelmélet* (systems theory), matematikai eszközökkel ír le rendkívül speciális rendszerek kvantitatív egymásrahatásait. A másik szélsőséges ág, a *rendszer-szemlélet* (systems thinking) a rendszereken alapuló megközelítést globálisan, gyakran meghökkenítően szerteágazó problémákra (pl. csoport-pszichológia) kívánja felhasználni, azonban esetenként a matematikai precizitás rovására.

Az információ reprezentációjának témaköre egy zavaróan szerteágazó családfához vezetett el bennünket, ami inkább elbizonytalaníthatja, mint segíti kérdésfeltevéseinket. Mindezek, a felhasznált fogalmaikban rokon tudományágak ugyanis nem, vagy csak alig képesek leírni a számítógépes rendszerek fejlesztésekor felmerülő problémákat. Holott a fejlesztés során a

valóságnak a rendszerrel kapcsolatos részeiről információkat kell szerez-nünk, azokat modelleznünk kell, a rendszernek a követelményekkel megfo-galmazható elvi felépítését és elvi működését pedig reprezentálnunk kell a számítógépen.

Az informatikát, azon belül is az információs rendszerek fejlesztését te-kinthetjük egy olyan katalizátornak, amely problémáival, illetve eszközkész-letével segít kialakítani egy tiszta fogalomrendszert az információ ábrá-zolásáról, illetve az ismeretszerzésről. A számítógép így teremti meg azt az alkímiai laboratóriumot, amely a pszichénél összehasonlíthatatlanul egysze-rűbb felépítésével, a reprezentáció megtekinthető formájával, a rendkívül primitív tároló és műveleti alapeszközökkel, a megfogalmazások (programok és adatok) nagy bonyolultságának, ugyanakkor tökéletes pontosságának az igényével lehetővé teszi, hogy az információ reprezentációival „vegytiszta” körülmények között szembesüljünk. Ezek a körülmények hasonlíthatók a telekommunikáció által megteremtett „vegytisztsághoz”, amely az átviteli, azaz a mennyiségi oldalról közelítő információelméletet megteremtette.

Elsődleges, „taktikai” célunk így egy olyan fogalomrendszer kialakítása, amellyel az információs rendszerek fejlesztése matematikai egzaktsággal, ugyanakkor a lehető legkevesebb eszközkészlet (halmazok, függvények, stb.) felhasználásával leírható. Mindezzel azonban csupán a célterületet határoltuk be, ugyanis a szemlélet azonnal alkalmazható *általában* az infor-mációk leírására és *általában* az ismeretszerzésre. A fogalmak elnevezései azonosak a korábban említett tudományágakban használt elnevezésekkel, azonban szinte mindig kénytelenek vagyunk eltérni az ott alkalmazott értel-mezésektől. Az eltérések ugyanakkor rokoníthatók a többi értelmezéssel, ami sajnos egy meglehetősen zavaró helyzetet teremt, de a matematikai definí-ciók remélhetőleg segítenek a tisztázásban. A fogalmak eltérő értelmezése miatt ezért nem is tekintjük témánkat az említett tudományágakhoz szorosan kapcsolódónak!

A következő oldalak célja tehát egy olyan elvi megközelítés felvázolása, amely egyszer elvezethet a rendszerfejlesztés egzakt fogalmi kereteinek ki-alakításához, valamint az általános ismeretszerzés és reprezentáció legalap-vetőbb fogalmainak matematikai precizitású tisztázásához.

Objektumorientált szemlélet, OMT, UML

Az információs rendszerek fejlesztésében az egyik legmodernebbnek tekintett irányzat az *objektumorientált módszer*, mely több évtizedes, kísérletinek tekinthető megoldások után az utolsó években vált kiforrottá, széleskörűen elterjedté és gyakorlatilag egyeduralkodóvá. Mi is ez a szemlélet?

A korábbi módszerek elkülönítve kezelték a tárolandó adatokat az azokat kezelő műveletektől. Az objektumorientált megközelítésben a rendszert olyan objektumokként adjuk meg, amelyben együtt szerepelnek az objektumot leíró tulajdonságok (a tárolandó adatok), az azokat kezelő műveletekkel. Az objektum így egy olyan önálló egység, amely „ismeri” a saját adatait és működését. A kétfajta programelem egy helyen és együttesen történő megadása miatt azok sokkal szervezettebben kapcsolhatók össze. A beírható és visszaolvasható adatsomagok helyett így olyan programcsomópontokat definiálhatunk, amelyek sokkal változatosabb módon „szólíthatók” meg, azaz a primitív tárolás/visszatöltés lehetőségén túllépve egy *egységes műveleti potencialitás* változatosságaként, azaz a végrehajtható műveletek gyűjteményeként jelennek meg. Az objektumorientált eszközrendszer igazi jelentősége, hogy a végrehajtható műveletek felsorolásával az objektumok programegységei absztrakt módon fogalmazhatók meg, elrejtve azok megvalósításának módját, mely absztrakt képek már részben elszakadhatnak a konkrét számítógépes fogalomrendszertől. Mindez azt jelenti, hogy már nem csak a tervek, hanem a programok szintjén is lehetséges az információk manipulációjának egy elvi megközelítési módja.

Érdekesség, hogy bizonyos objektumorientált megoldások már a programozás kezdeti időszakában megjelentek egyes korai mesterséges intelligencia rendszerekben. Az első, ténylegesen objektumorientáltnak nevezhető, egységes eszközrendszerű programozási nyelv a '67-ben megszülető SIMULA volt, melyet – ahogy a neve is elárulja – alapvetően szimulációs feladatok megoldására terveztek. Az „objektumorientált” szókapcsolatot először a '70-es években kifejlesztett Smalltalk nyelv, illetve rendszer jellemzésére használták, amely szemlélete már a hagyományos programszervezési módoktól is eltér: itt a programot az objektumok közötti üzenetváltásoként kell megvalósítanunk. A fejlődő eszközrendszer szoftver-ipari felhasználásának kapuját a korábban rendkívül népszerű C nyelv objektumorientált kiegészítése, a '80-as évek elején megjelenő C++ nyitotta meg. A '90-es évek

közepén született meg az „Internet nyelve”, a C++ jelöléseit alapul vevő, de sokkal egyszerűbb és áttekinthetőbb fogalmakra épülő Java programozási nyelv, illetve a hasonló koncepciókra épülő C# nyelv, melyekkel az objektumorientált eszközrendszer könnyebben kezelhetővé vált, és amely így hamarosan széleskörű elterjedtséget eredményezett.

A programozási nyelvek mellett az objektumorientált eszközök hamarosan a rendszerfejlesztési módszerekben is megjelentek. Először a korábbi szervezési módszerek bővültek objektumorientált kiegészítésekkel, majd tucatjával születtek újabb és újabb, már tisztán objektumorientált eszközökre épülő módszertanok. A kisebb-nagyobb népszerűséget elért módszerek között különleges helyet foglal el a GE kutató laboratóriumában, a Rumbaugh, Blaha, Premierlani, Eddy és Lorensen alkotta csoport által összeállított Objektum Modellezési Technika (OMT). Az OMT a korábbi objektumorientált módszertanok szintézise, melyben a leggyakoribb jelölésekből kiindulva sikerült egy hatékonyan alkalmazható, konzisztens jelölésrendszert összeállítani. Az OMT rövid idő alatt az egyik legnépszerűbb módszerre vált, mely népszerűség részben az OMT-t ismertető könyv közérthető, mélyebb informatikai ismeretek nélkül is elsajátítható voltára vezethető vissza.

Az OMT alapcélkitűzése a számítógépes fogalmaktól mentes elvi megközelítés, absztrakt leírás. Szlogenje szerint a fejlesztésben a feladat, a „*mit*” az elsődleges, nem pedig annak konkrét program-implementációs módja, a „*hogyan*”.

Az információ reprezentációjának témája szempontjából az OMT különleges fontosságú, ugyanis itt minden fogalmat, igaz csak informálisan (szövegesen), de egyszerűen és érthetően definiáltak. Az alapvetően a szemlélet oldaláról közelítő OMT definíciói, illetve a könyvben fellelhető néhány állítás már kijelöli azt a fogalmi célterületet, amelyet az ismeret-reprezentáció vizsgálatakor le kell fednünk. Témánk így részben tekinthető az OMT informális definícióinak és állításainak egzakt, matematikai formába való átültetéseinek, illetve pontosításainak.

Az OMT megjelenése óta eltelt évek az objektumorientáltság szoftver-nagypari térhódítását eredményezték, szemléleti oldalról azonban sajnos csak kevés újdonságot hoztak. Ez a folyamat magán az OMT-n is megfigyelhető. A három legnépszerűbb módszertan megalkotója, Booch, az OMT-

csoportot vezető Rumbaugh, valamint Jacobson az egységesítés érdekében összeállította az OMT utódjának is tekinthető, azonban alapvetően ipari-technikai szemléletű Unified Modeling Language (UML) jelölésrendszerét, amely hamarosan egyszerre egyeduralkodóvá és szabvánnyá is vált.

Elméleti megközelítés

Milyen gyakorlati előnyökkel járhat egy elméleti megközelítés kidolgozása a rendszerfejlesztés, illetve az informatika témaköreiben, mely utóbbit most a *rendszerekre vonatkozó információk megszerzéseként és ábrázolásaként* definiáljuk, melybe a programrendszerként, más néven „alkalmazásként” történő reprezentációt is beleértjük.

Egy kis túlzással azt mondhatnánk, hogy egészen a legutóbbi évekig a rendszerfejlesztést csak kísérletezésnek lehetett tekinteni, amelyet esetenként célszerűbb lett volna a külvilágtól elzárt laboratóriumokban, nem pedig valós körülmények között végrehajtani. Egészen egyszerűen a számítógépek megjelenése óta eltelt néhány évtized még nem volt elegendő ahhoz, hogy a rendszerfejlesztésnek kialakuljon a technológiája. A technológia még most sem teljes, azonban napjainkra már összegyűlt annyi gyakorlati tapasztalat, amely elegendő az átlagos feladatok megoldásához. Az átlagtól eltérő helyzetek, a megszokottnál csak egy kissé nagyobb bonyolultság azonban még most is problémákat okoz. Ami a technológia kialakulatlansága ellenére mégis kikényszerítette a valós körülmények között működő programrendszerek fejlesztését, az a számítógép rendkívüli lehetőségeinek felismerése volt. Az összetett rendszerek iránti igény napjainkra a többszörösére nőtt, mind mennyiség, mind bonyolultság szempontjából.

A rendszerfejlesztés eddig többnyire a próbálkozás útján fejlődött és a sikeres kísérletek tapasztalatai tették lehetővé az egyre bonyolultabb belső struktúrájú alkalmazások elkészítését és így az újabb kísérletezéseket. Egy megfelelő elméleti-szemléleti háttér segítségével célzott irányba történő, és ezért gyorsabb ütemű fejlődés érhető el.

Az informatika eddigi története alapján a fejlődésnek valószínűsíthető néhány karakterisztikus iránya. Például az eddigi programreprezentációs jelölések – így a programnyelvek és a grafikus eszközök is – egyre inkább elszakadnak a számítógép fogalmaitól és mind közelebb kerülnek a valósághoz. Képletesen: a valóság fogalmai és viszonyai beköltöznek a számítógép, alapvetően csak primitív adattároló és számítás-alapú szemléletének a vilá-

gába. Az ismeretszerzés és reprezentáció fogalmainak tisztázásával újabb lépéseket tehetünk ezen az úton, mely eredményezheti például egy informatikai (nem pedig számítástechnikai) nyelv és jelölésrendszer kialakulását, amely automatikusan programmá fordítható; már jelenleg is létezik több ilyen, jelenleg még csak részleges alkalmazhatóságú kísérlet. Hasonlóan, elképzelhető, hogy a valóság elemeinek, azok viszonyainak és egymásra hatásának szemlélete alapján a számítógépek egy új generációja, – a „számítógép” elnevezés mintájára – egy „információgép” készíthető el, amelyben a valós folyamatokat nem primitív számítási-tárolási műveletekként, hanem a kiindulási leíráshoz közel álló, magasszintű elemekkel és műveletekkel adhatjuk meg.

Az objektumorientáltság megjelenésével nem zárult le a fejlődés. Napjainkban is több olyan kísérleti módszer van, amely túl kíván lépni az ott meghatározott kereteken. Egy elméleti alapokon nyugvó megközelítéssel kijavíthatók a jelenlegi szemléletek hibái és következtelenségei, kiegészíthetők a hiányosságok, illetve az elvek elvezethetnek egy új rendszerfejlesztési módszerhez.

Fogalmak és szemlélet

A későbbi formális definíciók előtt először szövegesen áttekintjük elméleti megközelítésünk legfontosabb fogalmait. Legalapvetőbb fogalmunk a rendszer, amely egyetlen megfigyelhető tulajdonságának a komplex állapotát és annak változását tekintjük. Az elméleti megközelítés eredményei a valóságra is alkalmazhatók, amennyiben annak valóban sajátja néhány feltételezésünk, melyeket hipotézisekként sorolunk fel.

Szemlélet

Célunk az információ reprezentációjának vizsgálata, az információ azonban mindig valamire, általában valamely önmagán túli dologra vonatkozik. Ezt a legalapvetőbb dolgot *rendszernek* nevezzük, mely egyetlen megfogható és leírható jellegzetességének a *változását* tekintjük, melyet egy absztrakt külső szemlélő szempontjából vizsgálunk. A változást a rendszer *állapotának* nevezett, együttesen vett tulajdonságainak a változásaként értelmezzük. Legalapvetőbb fogalmunk tehát a rendszer (az alany), amelyet a változásával (az állítmánnyal, azaz a rá vonatkozó állítással) írunk le, amelyet így valójában *folyamatként* (alany és állítmány együttesen) tekintünk.

Maga a *rendszer* fogalmából természetes módon a rendszereken alapuló szemléletre asszociálnánk, azonban, ahogy azt korábban is hangsúlyoztuk, az azonos elnevezések ellenére most egy más kiindulópontot kell választanunk.

A rendszereken alapuló szemlélet a rendszert, az attól külön választott környezetével való kapcsolataként és egymásrahatásaként tárgyalja. Megközelítésünkben azonban a rendszert *teljességében* vizsgáljuk, azaz a rendszeren kívül nincs semmi (az időn, képletesen: a megfigyelő idején kívül, valamint a megfigyelhetőség tartományán, azaz az állapothalmazon kívül), nincs környezet és nincs külvilág. A hagyományos „rendszer, környezete és külvilága” megközelítéssel ellentétben itt ezek együttesen alkotják a rendszert, illetve ezek mind a teljes rendszer részei. Ebben a szemléletben a „környe-

zet” és „külvilág” fogalmak a rendszer egy-egy részének a kiválasztása esetén értelmezhetők.

Adott időpillanatban a rendszer részei bizonyos állapotokban lehetnek, ezért a teljes rendszer *komplex állapotát* az összes alkotóelem együttes állapota határozza meg. Ahogy a rész, környezet és külvilág együttesen, egyetlen rendszerként történő vizsgálata is egy egységesebb megközelítést tesz lehetővé, hasonlóan, az együttes állapot alapul választása is leegyszerűsíti a fogalomrendszer felépítését. Természetesen a későbbiekben az állapotkomponensek leírására is nyújtunk eszközöket.

A rendszereken alapuló megközelítésekben a változást általában lépésekben írják le: adott időpontban a (rész-)rendszer állapota és a környezet állapota esetén a *következő* időpontban ezek milyen állapotokba fognak válni. A rákövetkező állapotokon, azaz az *állapotváltásokon* alapuló megközelítés azonban rendkívül megnehezíti a rendszer átalakításait, ezért kénytelenek vagyunk egy ennél általánosabb kiindulópontot választani. A rendszert állapotváltások helyett *állapotváltozásként* írjuk le, amely azt jelenti, hogy minden időpontban megadjuk a megfelelő állapotot, azaz az állapot az időpontokon értelmezett függvény lesz. Ennek az általánosabb megközelítésnek egy speciális eseteként természetesen az állapotváltások is leírhatók.

A hagyományos megközelítésektől a negyedik, egyben utolsó eltérésünk, hogy azonos módon tárgyaljuk a tényleges rendszert és annak *megfigyelt* változatait. A megfigyelés egyrészt a rendszer idejét fogja jelenteni egy, a rendszeren kívüli fogalomként, másrészt pedig azt a bizonytalanságot, hogy a megfigyelő nem feltétlenül tudja megállapítani, hogy adott időpontban a rendszer pontosan milyen állapotban van, hanem csak az állapotok valamely halmazát tudja megadni. A másik, rendszeren kívüli fogalom így a rendszer által felvehető állapotok halmaza lesz, amely valójában a megfigyelő által azonosítható állapotokat fogja jelenteni. Az általánosabb időszemlélet mellett a bizonytalanság beépítése rendkívül megkönnyíti a fogalomrendszer felépítését. Például egy rendszer általánosabb változata esetén adott időpontokban bővíülhetnek az éppen lehetséges állapotok, speciálisabb változat esetén pedig szűkülhetnek.

A tárgyalni kívánt témák miatt kénytelenek vagyunk eltérni más megközelítések (kibernetika, rendszerelmélet, stb.) hagyományos kiindulópontjaitól. Az eltérések egy egységesebb és általánosabb megközelítést tesznek lehetővé, azonban ez az általánosság sajnos megnehezíti a definíciók értelmezését, amiért előre is kérjük a tisztelt Olvasó türelmét.

A későbbi egzakt, formális definíciók előtt most informálisan (szövegesen) összefoglaljuk megközelítésünk legalapvetőbb szemléleti kereteit.

Rendszernek egyszerűen egy *folyamatot*, például valamilyen változást nevezünk, mivel úgy tekintjük, hogy a változás az a legalapvetőbb dolog, amivel a rendszert le tudjuk írni. Egy folyamatot úgy adunk meg, mint bizonyos *tulajdonságok* módosulását, változását, mely tulajdonságokat együttesen a rendszer *állapotának* nevezzük.

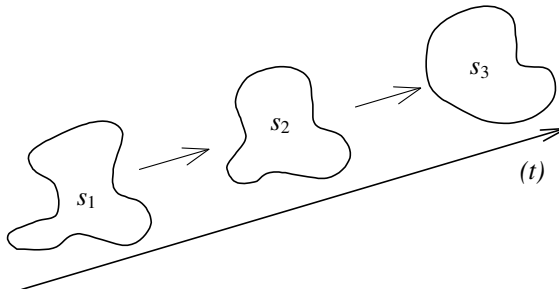
Egyes rendszereknél az állapot változása rendelkezik bizonyos egyediséggel és irányultsággal. Ekkor a rendszer által felvett állapotok különböznek egymástól, illetve bármely két különböző állapot esetén megállapítható, hogy melyik a „korábbi” és melyik a „későbbi” állapot. Az egyediség és irányultság például valamely érték halmazódásával vagy csökkenésével szemléltethető, míg matematikailag az ezen értékek halmazának rendezettségével írható le. Ha a rendszerből képezhető egy ilyen szigorúan monoton módon változó érték, akkor azokat *időpontoknak*, azok halmazát pedig *időnek* nevezzük.

Ha az állapot nem változik vagy a változása ciklikus, akkor ezt az állandóságot, illetve ismétlődést lehetetlen a rendszeren „belülről” (azaz csak a rendszer állapotának ismeretében) megfigyelni. Az ismétlődés (illetve annak speciális esete, az állandóság) csak egy olyan külső megfigyelő által észlelhető, aki rendelkezik egy szigorúan monoton módon változó értékkel, azaz idővel. A megfigyelt rendszer így egy olyan nagyobb rendszer része, amely a megfigyelőt is magában foglalja. Rendszeren mindig „észlelt”, „megfigyelt” rendszert értünk, azaz a rendszer jelentheti valamely más rendszer „megfigyelt” képét, mely speciális esetben azonos lehet magával a (megfigyelt) rendszerrel. A megfigyelőt azonban csak az általa azonosítani képes állapotok halmazára, valamint ezen megfigyelő idejére egyszerűsítjük le és a rendszert ebbe az időbe helyezzük el, valamint ezen állapotokkal írjuk le. Természetesen a rendszerben is lehet idő, azaz az állapotnak egy halmozódó vagy csökkenő érték-komponense.

Rendszeren tehát egy időben lezajló folyamatot értünk, amely azt jelenti, hogy a rendszer minden időpontban egy-egy állapotban van. Az idő a megfigyelő idejét jelenti, ezért lehetséges, hogy a rendszer különböző időpontjaihoz ugyanaz az állapot tartozik. Ez nem nevezhető tényleges változásnak, ezért is használtuk először a változás helyett a bővebb értelmű „folyamat”

szót. Ugyanakkor a *változás* mégis tekinthető a folyamat szinonimájaként. Egyrészt, mivel az állandóság, illetve a ciklikusság tekinthetők a változás speciális eseteinek, másrészt, mert a folyamat időben zajlik le, azaz rendelkezik egy szigorúan monoton módon változó értékkel.

A változás ideje a megfigyelő idejét jelenti. A rendszer minden időpontban valamilyen állapotban van, de lehetséges, hogy (mi megfigyelők) azt nem ismerjük vagy attól eltekintünk. A megfigyelt folyamat – azaz a rendszer – matematikailag egy olyan függvényként definiálható, amely az időpontok halmazának egy részhalmazát, a „megfigyelés időpontjait” az állapotok halmazába képi. A rendszer tehát egy függvény, így a „megfigyelés időpontjainak” halmazát a *rendszer idejének* vagy a rendszer által *értelmezett időpontok* halmazának nevezzük.



1. ábra. Rendszer: állapot változása az időben

Egy változás nem feltétlenül homogén: lehetséges, hogy a tulajdonságok felbonthatók megkülönböztethető *komponensekre, részekre*. (A „rész” szót a komponens szinonimájaként használjuk.) A rendszer **részekre történő felbontását** a rendszer komponenseinek megkülönböztetéseként, a rendszer komponensét pedig informálisan relatív, azaz eltérő (nem homogén) változásként szemléltethetjük. Egy rész így más részek változása melletti eltérő változásként észlelhető, például változáson belüli viszonylagos állandósággént, vagy állandóságon belüli viszonylagos változásként.

A változásban előfordulhat ismétlődés, azaz lehetséges egy komponenst meghatározó részállapottal azonos vagy ahhoz hasonló jellegű részállapot más időpontban vagy más „helyen” történő előfordulása, amely azonos vagy hasonló jellegű változást eredményezhet. A hasonló változást eredményező

konkrét komponensek együtteseként írhatjuk le informálisan a (rész)rendszerek **absztrakcióját**. A *részek hasonlósága* a következőképpen szemléltethető: egy rendszer részletét minél hasonlóbbal helyettesítjük, annál kevésbé (például kevesebb további rész esetén) fog az eredeti folyamattól eltérni a változás.

Példa: Állandóságon belüli viszonylagos változás.

Egy hagyományos óra számlapja tekinthető a mutatók viszonylagos változásának állandó háttereként. A mutatók így a rendszer önálló komponenseiként észlelhetők. Ugyanakkor egy, az egyik mutatón elhelyezkedő megfigyelő számára maga a mutató jelenik meg a környezet változásán belüli viszonylagos állandóságként.

Példa: Absztrakció és hasonlóság.

Két működő, azonos típusú hagyományos óra esetén az egyik percmutatóját a másik percmutatójával helyettesítve a rendszerünk továbbra is azonos módon változik. A percmutatót egy másik típusú óra percmutatójával helyettesítve az eredeti rendszerhez hasonló változást kapunk, így a mutató mozgásának jellege – adott tengely körüli, adott sebességű forgás – azonos marad. A hagyományos órák percmutatói tehát hasonlóak, de szintén (gyengébb) hasonlóságot állapíthatunk meg például a percmutatók és a kismutatók között.

Hipotézisek

Az informatika feladatát a *valóságon alapuló* rendszerek megismeréseként és modellezéseként határozhatjuk meg.

A definícióival és állításaival a következőkben ismertett fogalomrendszer természetesen egy zárt világot alkot, amelyről – mint elméletről –, annak helyességéről, következetes voltáról önmagán belül kell véleményyt alkotni.

Ha annak a lehetőségét is meg szeretnénk teremteni, hogy a fogalomrendszert valós rendszerek felbontására és ábrázolására, tehát közvetve a valóságra is vonatkoztathassuk, akkor az már *szemléleti* kérdés, amelyhez rögzítenünk kell bizonyos **hipotéziseket**, melyek feltételezésünk szerint a valóság jellemzői. A „valóság” megjelölést az *univerzum*, azaz a minket körülvevő egyetemes világ szinonimájaként használjuk. Ha a későbbiekben annak „valós” jellegét kívánjuk hangsúlyozni, akkor arra a „valós világ” szókapcsolattal utalunk.

1. **A valóságra alkalmazható a rendszereken alapuló szemlélet.** Azaz a valóság tekinthető változásként, amely az – együttesen állapotnak nevezett – tulajdonságainak az időben lezajló módosulását jelenti.

Feltételezzük, hogy a valóság változása egyediséggel és irányultsággal – azaz **idővel** is rendelkezik. Valós világ esetén ezt az irányultságot egy fizikai törvénnyel (energiaköltség nélküli rendszer átlagos entrópiájának növekedésével) szemléltetik.

Feltételezzük, hogy a rendszereken alapuló szemlélet **teljes**. Ezen azt értjük, hogy a valóságnál nincs bővebb rendszer, azaz a valóság nem valamely más rendszer absztrakciója. Más szóval: a valóságnak nincs olyan aspektusa, amely a szemlélettel nem ragadható meg. A teljesség egyik következménye, hogy minden ismeretünk végső forrása a valóság. Másik következmény, hogy a valóság ideje ($\mathcal{T}_U$ – univerzális idő) egy olyan, lehetséges legnagyobb számosságú halmaz, amely egy-egy monoton növekvő bijektív képnek részhalmaza minden rendszer ideje.

Feltételezzük, hogy a valóság minden időpillanatban valamilyen állapotban van, legfeljebb azt nem ismerjük, vagy attól eltekintünk. A valóság modellezhetősége szükségessé teszi annak feltételezését is, hogy a valóság adott időpillanatban felvett állapota **megfigyelhető**, tehát rendelkezik bizonyos objektivitással. A „bizonyos” szó azt jelenti, hogy a megfigyelők is hathatnak – akár magával a megfigyelés tényével – a valóságra.

2. **A valóság részekre bontható**, azaz változása nem feltétlenül homogén, a tulajdonságai megkülönböztethetők. Ez azt jelenti, hogy értelmezhető a valóságon alapuló rendszerek felbontása.

3. **A valóságban előfordul ismétlődés**, azaz lehetséges egy részt meghatározó tulajdonság-kombinációval azonos vagy hasonló jellegű kombináció más időpontban vagy más „helyen” történő előfordulása, amely azonos vagy hasonló jellegű változást eredményezhet. Tehát értelmezhető a valóság, illetve egyes részeinek az absztrakciója („absztrakt fogalomalkotás”).

Ezek a filozófiába hajló hipotézisek hétköznapi megfigyeléseink szerint igazak: például a valóság monoton és egyedi módon változik. A hipotéziseket azonban *mindössze (!)* a megjegyzésekben és a példákban használjuk fel. A következő eredmények a hipotézisek nélkül is érvényesek, például egy számítógépes környezetben. A hipotézisek segítségével azonban az *elmélet*

eredményei – mint *szemlélet*, hasonlóan a rendszerszemlélethez – a valóságon alapuló rendszerek megközelítésére is alkalmazhatók.

Definícióink kimondva explicit, vagy kimondatlanul implicit módon mind rendszerekre vonatkoznak, mint például rendszerek absztrakciója, (rendszerek) pontosítása, (rendszerek) felbontása, (rendszerek) korlátozója... A hipotézisekben rögzített feltételezéseink szerint a valóságra alkalmazható a rendszereken alapuló szemlélet, ezért *feltételezzük*, hogy a rendszerekkel kapcsolatos definíciók egyben a valóságra vonatkozó definíciók is. Ezért van egy közvetett hipotézisünk is: feltételezzük, hogy a rendszerekkel kapcsolatos definíciók a valóságra alkalmazva a köznapi megnevezéseknél a fogalmakat se a szükségesnél nem jobban, sem az elégségesnél kevésbé nem korlátozzák.

Alkalmazott jelölések

Az ismeretszerzés és -reprezentáció témaköre nem a matematika egy részterülete, bár a precizitás miatt a megfogalmazásokban természetesen a matematika eszközkészletét használjuk fel. Mivel a gondolatmenet ismertetésekor alapvetően egy szemléleti-filozófiai megközelítést követünk, valamint, hogy a nem matematikus olvasók számára is áttekinthető legyen a szöveg, eltérünk a matematikai eredmények ismertetésekor használt definíció-tétel jelölésektől. Kérjük ezért a matematikus olvasókat, hogy fogadják el ezt az engedményt a nem-matematikusok kedvéért.

Megközelítésünk alapvetően megnevezni, definiálni kívánja a fogalmakat és azok viszonyait, ezért a definíciók számához képest viszonylag kevés a nagyobb lélegzetű tétel; azok elsősorban a reprezentációval kapcsolatosak. A triviális állítások bizonyítását mellőzzük, a nem triviális eseteknél pedig mindössze a vázlatos felépítést adjuk meg.

A következő formátumkonvenciókat alkalmazzuk:

- A *definíciókat* a ∇ jel vezeti be. A definíción belül *dőlt betű* emeli ki a definiált fogalmak elnevezéseit. A közbevetett megjegyzések és kiegészítő tulajdonságok zárójelek között szerepelnek.
- Az *állításokat* a § jel és vastag betűs cím vezeti be, mely cím az állítás összefoglaló elnevezése.
- A *megjegyzéseket*, a ☞ jel és dőlt betűs mondat jelöli.

A gondolatmenet leírása általában a matematikában szokásos jelölésekkel történik, a következő eltérésekkel:

- Az „akkor és csak akkor, ha” állításokat a „*pontosan akkor, ha*” szókapcsolat írja le.
- Természetes számnak a 0 számot és rákövetkezőit tekintjük.
- D_f jelöli az $f : A \rightarrow B$ függvény értékkészletét, azaz B -t.
- Az $f : A \rightarrow B$ függvény, illetve $A' \subseteq A$ esetén az $f(A')$ a B azon értékeit jelöli melyekre $\{b \in B \mid \exists a \in A' : f(a) = b\}$. A függvény ekkor az A' kiindulási halmaz alapján „kiválasztja” a B értékkészlet $f(A') \subseteq B$ részét.
- Az $a_{i \in I}$ az egyszerűsített jelölése az indexelt a_i elemeknek, valamint egy I indexhalmaznak, ahol az i sorra felveszi az I értékeit. Műveletek, pl. $\bigcup_{i \in I}$ esetén a hagyományos értelmezést használjuk, tehát ebben a példában: „ i sorra felveszi I indexhalmaz elemeit”.



Absztrakció és kompozíció

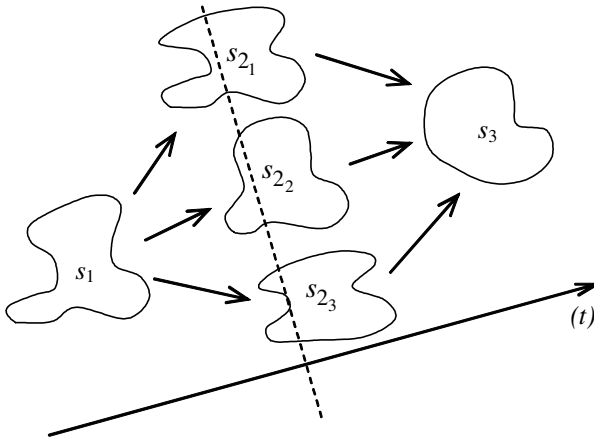
Legalapvetőbb fogalmunk a rendszer. Egy rendszert különböző módon is megadhatunk, így adott vizsgálatot leegyszerűsítethetünk a megfelelő forma kiválasztásával. A megfelelő állapothalmazú rendszer a komponenseire bontható. Bizonyos rendszerek hasonlíthatók, s ezek körében definiálható egy feltétel/következmény fogalom. A változások változatossága összehasonlítható az ekvivalencia és az absztrakciók eszközeivel.

Rendszer

Vizsgálataink legalapvetőbb tárgyát rendszernek nevezzük, melyet teljességében vizsgálunk, azaz a rendszeren kívül nincs semmi (az időn, képletesen: a megfigyelő idején kívül, valamint az azonosítható állapotokon kívül), nincs külvilág. Egy adott időpillanatban a rendszer részei adott állapotokban lehetnek, ezért a teljes rendszer komplex állapotát az összes alkotóelem együttes állapota határozza meg.

A rendszerrel kapcsolatban az egyetlen megfogható dolognak a változást tekintjük, ezért a *rendszer* és a *változás* szavakat szinonimaként tekintjük. Az *idő* (vagy *ütem*) a változás érzékeléséhez és ábrázolásához szükséges egymásutániságot írja le. Mivel az idő egyértelműen meghatározza a rendszer által felvehető állapotokat, ezért ennek függvényében adható meg a rendszer változása, a folyamat. Bár vizsgálatunk során az időt kiemeljük, lehetséges, hogy a komplex állapotban létezik erre utaló komponens.

Determinisztikus megadás esetén a rendszer egy adott időpontban pontosan egy állapotban lehet. Általános esetben azonban megengedjük, hogy egy időpillanathoz ne csak egy állapotot, hanem egy állapot-kombinációt, a lehetséges állapotok halmazát rendeljük. Ez az általános vagy nemdeterminisztikus megadási mód lehetővé teszi, hogy a változás több lehetséges lefutását is egyetlen rendszerként ábrázoljuk.



2. ábra. Általános megadás esetén a rendszernek egy időpontban több állapota is lehetséges

Tekintsük tehát első és legalapvetőbb definíciónkat:

▼ **Rendszer.** Formálisan egy σ (általános vagy nemdeterminisztikus megadású) *rendszer* egy $(\mathcal{T}, S, P)$ elemhármas,

$$\sigma = (\mathcal{T}, S, P)$$

ahol $\mathcal{T}$ nem üres, jólrendezett halmaz a rendszer változásának ideje, S nem üres halmaz a rendszer lehetséges állapotainak halmaza, $P: \mathcal{T} \rightarrow 2^S \setminus \{\emptyset\}$ *állapotfüggvény* (vagy a *változás függvénye*) pedig az adott időpontban felvehető állapotok halmazát, így a *rendszer változását*, a *folyamatot* határozza meg.

Feltevésünk szerint a rendszer minden időpillanat esetén valamilyen állapotban van, ezért nem engedjük meg, hogy a változás függvénye bármely időponthoz „értelmezhetetlen” állapotként üreshalmazt rendeljen.

▼ **Tetszőleges vagy értelmezetlen állapot, tetszőleges absztrakt rendszer.** Azokban az időpillanatokban, amelyekhez az állapotfüggvény a teljes állapothalmazt rendeli ($P(t)=S$) azt mondjuk, hogy a rendszer *bármilyen, tetszőleges, bizonytalan, értelmezetlen* (de nem értelmezhetetlen) vagy

meghatározatlan állapotban van. Szélső helyzetben, ha a rendszer minden időpillanatban meghatározatlan állapotú, akkor egy-egy (különböző $\mathcal{T}$ és S halmazú), minden időpontban értelmezetlen rendszert kapunk, amelyet *tetszőleges absztrakt rendszernek* (jelölés: $\mathfrak{R}_{(\mathcal{T}, S)}$) nevezzük.

▼ **Értelmezett időpontok, értelmezett állapotkombinációk.** Azokat az időpillanatokat, amelyekhez a változás függvénye nem a teljes állapot-halmazt rendeli, *tényleges*, vagy a rendszer által *értelmezett időpontoknak* nevezzük és ezek halmazát T , vagy a kiemelést hangsúlyozva $\bar{T}$ módon jelöljük. A lehetséges állapotok S halmazából az értelmezett időpontokban az *értelmezett*, más néven *ténylegesen felvett állapot-kombinációk* halmazára $(P(T) \subseteq 2^S \setminus \{\emptyset, S\})$ az $\bar{S}$ jelölést használjuk.

Példa: Rendszer.

Egy tó vizének hőmérsékletét egy rögzített időpontban és egy adott ponton mérjük meg és jegyezzük fel egy adott év minden napján. Rendszerünk 365 vagy 366 értelmezett időponttal rendelkezik, míg a többi időpillanatban értelmezetlen, bizonytalan az állapota.

Másik példaként választott rendszerünk legyen az óra-számjegyek értéke egy olyan digitális órán, amely pontos, de egy hiba miatt véletlenszerűen vált a 12/24 órás kijelzés között. Ha eltekintünk az éppen aktuális kijelzési módoktól (mely vagy a 12 vagy a 24 órás kijelzés lehet), akkor a 13 és 0 óra közötti napszakban az órán megjelenő számértéket egy-egy kételemű halmazzal, mint két lehetséges értékkel ($\{1, 13\}$, $\{2, 14\}$, ...) adhatjuk meg. Tehát délután 3 és 4 óra között vagy a 3-as vagy a 15-ös számjegy jelenik meg.

▼ **Rendszer normalizált megadása.** Rendszert *normalizált módon* a $\sigma = (T, T, S, P)$ elemnégyessel adunk meg, ahol $T \subseteq \mathcal{T}$ az értelmezett időpontok halmaza és az állapotfüggvény $P: T \rightarrow 2^S \setminus \{\emptyset, S\}$ alakú.

A két megadási mód egyértelműen egymásba alakítható, ezért azokat egyenértékűeknek tekintjük.

A T halmaz újabb értelmezetlen időpontokkal történő bővítése – bár más rendszert eredményez – lényegében nem módosítja a rendszert.

Rendszer egyszerűsített megadásának (T, S, P) elemhármasa csak egy ekvivalencia osztály erejéig adja meg a változást, függetlenül attól, hogy egy konkrét rendszer mely T -nél bővebb $\mathcal{T}$ időbe ágyazza azt.

✓ **Rendszer egyszerűsített megadása.** A $\sigma = (T, S, P)$ elemhármassal történő egyszerűsített megadás a (normalizált módon megadott) különböző T_i időbe ágyazott (T_i, T, S, P) rendszerek egy (tetszőleges) reprezentáns-rendszerét jelöli.

Rendszer egyszerűsített megadása esetén így a változás keretétől szolgáló, de a $T \setminus T$ -n kívül csak értelmetlen időpontokat tartalmazó T helyett kizárólag az értelmezett időpontokat tekintjük. Képletesen: a normalizált megadás T halmazát tetszőleges, T -nél bővebb halmazra cserélhetjük.

Példa: Életjáték.

Egy Ulam-féle sejtautomata (közismert nevén „életjáték”) minden cellája „üres” vagy „élő” állapotban lehet. Egy $n \times m$ -es rács esetén az állapothalmaz így az $\{\text{üres}, \text{élő}\}$ kételemű halmaz nm -szeres Descartes-szorzata lesz, mely leírható a $[0; 2^{nm}-1]$ közötti (pl. bináris számjegyű) egészszámokkal. 5×4 -es rács esetén ez egy 0 és 1048575 közötti számjegyet jelent. Ha egyetlen időpontban egynél több elemű az állapotkombináció, akkor nem ismerjük pontosan a szituációt, mely értelmetlen állapot esetén az a maximális elemszámú, 1048576 elemből álló teljes halmaz.

Az időpontok T halmazának válasszuk most az N természetes számok halmazát, ahol 0 az életjáték indítási pontja. Ha bizonyos időpontokban nem tudjuk megfigyelni a sejteket („elfordítjuk a fejünk”), akkor ott értelmetlen állapotként a teljes S állapothalmazt kell megadnunk. *Normalizált megadás* esetén el kell hagynunk ezeket a változás függvényének megadásából. Az *egyszerűsített megadásban* kizárólag az értelmezett időpontok szerepelnek, így az jelöl olyan reprezentáns rendszert is, amelyben például az időpontok teljes halmaza az egész számok, vagy a racionális, vagy a valós számok... halmaza.

✓ **Statikus, dinamikus és szigorúan változó rendszer.** Az *állandó, statikus* vagy *konstans rendszer* állapotkombinációja minden $t \in T$ esetén azonos. *Változó* vagy *dinamikus rendszernek* van legalább két különböző értelmezett állapotkombinációja. Ha a $P_{|T|}$ invertálható (azaz az állapotkombinációkból számítható az idő), a rendszert *szigorúan változó*nak nevezzük.

✓ **Szituáció vagy helyzet.** Rendszer egyetlen időpontra történő szűkítése, azaz az egyetlen időpontban értelmezett állapotú rendszer maga is rendszer, amelyet szituációnak, helyzetnek vagy pillanatfelvételnak nevezünk.

☞ *Helyzet egyszerre statikus és szigorúan változó*, mivel szigorúan változó rendszer esetén mindössze a változás függvényének az invertálhatóságát követeltük meg.

✓ **Közvetlenül hasonlítható rendszerek.** A $\sigma_{i \in I} = (\mathcal{T}, S, P_i)$, tehát közös idő- és állapothalmazú rendszereket *közvetlenül hasonlítható rendszereknek* (jelölés: $\Sigma^{(\sigma_i)}$) nevezzük. $\Sigma^{(\sigma)}$ jelöli a σ rendszerrel közvetlenül hasonlítható rendszerek halmazát.

(Figyeljük meg a „többes-szám” jelét, a $\sigma_{i \in I}$ jelölésben, amely így több, az I indexhalmaz elemeivel indexelhető rendszert jelöl. A definíciókban ezeknek a rendszereknek csak az állapotfüggvénye tér el, így a σ_i rendszer állapotfüggvényét P_i jelöli.)

Természetesen minden rendszer önmagával közvetlenül hasonlítható, melyet nem szükséges külön jelöléssel hangsúlyoznunk. Ezt kihasználva a $\Sigma^{(\sigma_i)}$ több rendszerre vonatkozó tulajdonságot határoz meg, a $\Sigma^{(\sigma)}$ jelöléssel azonban ettől eltérő módon, adott rendszerek halmazát jelöljük.

A közvetlenül hasonlíthatóság azt jelenti, hogy adott időpontban a két rendszer által felvehető állapotok halmazát össze tudjuk hasonlítani, hogy pl. egyik a másiknak részhalmaza-e. Az időpontokban felvett állapotok összehasonlításával, így maguk a rendszerek is összehasonlíthatók.

✓ **Hasonlítható tetszőleges absztrakt rendszer.** Bármely σ rendszerhez megadható egy vele közvetlenül hasonlítható tetszőleges absztrakt rendszer, amelyet *hasonlítható tetszőleges absztrakt rendszernek* (jelölés: $\mathfrak{R}_\sigma$) nevezzünk.

☞ *A korábbi definíció alapján a tetszőleges absztrakt rendszer minden időpontjában bizonytalan állapotban van.*

Példa: Életjáték.

A természetes számok időhalmazú, illetve azonos állapothalmazú $n \times m$ -es életjátékok közvetlenül hasonlíthatók, azaz adott rendszerek esetén meg lehet állapítani, hogy bizonyos időpontokhoz a változás függvénye azonos, például értelmetlen állapotkombinációt rendel-e. Két rendszer

összehasonlításakor megállapíthatjuk, hogy minden időpontban az egyik állapothalmaza azonos vagy bővebb a másiknál; például adott időpontokban több állapotkombinációt is megenged. Ezen rendszerek körében a minden időpontjában a lehetséges legbővebb állapothalmazt megadó rendszer a hasonlítható tetszőleges absztrakt rendszer.

A rendszer definíciója alapvetően az időpontokban „felvehető állapotok” halmazzal történő megadásán alapszik. A rendszer fogalmát a halmazba tartozás egyértelműségének feloldásával bővíthetjük, például, ha az adott időpontban felvehető állapotokhoz valószínűségi értékeket rendelünk, jelen esetben azonban ez nem célunk.

Determinisztikus megadású rendszer

A rendszerek egy másik megközelítése esetén egy időpillanathoz nem a lehetséges állapotok halmazát, hanem egyetlen állapotot rendelünk.

Példa: $\{0,1\}^+$ jelsorozatot elfogadó DFA.

A táblázat egy determinisztikus véges állapotú automata (DFA) állapotváltásait ábrázolja az i input jelekből képzett 011210 sorozat hatására. Az ütemezést az egymás után következő szimbólumok valószínűsítják meg, így $T=N$ természetes számok. Az állapothalmaz az $S=\{0,1,2\} \times \{elf, elv\}$ Descartes-szorzat. Az átmeneti függvényt az ismert módokon, állapotdiagrammal vagy reguláris nyelvtannal határozhatjuk meg. Láthatjuk, hogy a 0 és 1 jelektől eltérő első 2 jel hatására az automata a következő időpillanatban az „elvet” állapotba megy át.

t idő	i input	s állapot
0	0	elf
1	1	elf
2	1	elf
3	2	elf
4	1	elv
5	0	elv
...	...	...

✓ **Determinisztikus megadású rendszer.** Rendszer $\sigma = (T, T, S, P)$ determinisztikus megadása esetén a változás $P: T \rightarrow S$ függvénye ($T \subseteq T$) az adott időpontban a rendszer által felvett állapotot adja meg. A $T \setminus T$ időpontokban a rendszer bizonytalan, értelmezetlen állapotú.

Determinisztikus esetben az $\bar{S} = P(T) \subseteq S$ jelölést a ténylegesen felvett állapotok (nem állapot-kombinációk) halmazának jelölésére alkalmazzuk.

Példa: Számítógép központi memóriájának tartalma.

Egy számítógép bekapcsolásától eltelt időpontokat megadhatjuk valós számokként, ahol a bekapcsolás előtti időpontokban bizonytalanak

tekintjük az állapotot. Ha minden időpontban megadjuk a központi memória tartalmát, akkor egy determinisztikus rendszert kapunk.

§ Rendszer megadható determinisztikus és nemdeterminisztikus módon.

A determinisztikus megadás bizonyos értelemben a normalizált nem-determinisztikus mód speciális esete. A transzformáció ebben az esetben mindössze annyit jelent, hogy a változás függvényében az eredeti rendszer állapota helyett az állapot egyelemű halmazát ($s \mapsto \{s\}$) írjuk.

A $\sigma = (\mathcal{T}, T, S, P)$ nemdeterminisztikus megadású rendszer esetén az új állapothalmaz legyen az $S' = 2^S \setminus \{\emptyset, S\}$ halmaz. A $P': T' \rightarrow S'$ (ahol $P'(t) = P(t)$ állapotfüggvényű rendszer már determinisztikus megadású lesz. Ezzel gyakorlatilag az állapotok nemdeterminisztikus (halmazzal megadott) lehetőségeinek *jelentésétől* eltekintünk és a halmazokat azok tényleges elemeitől elvonatkoztatva, mindössze *jelölésként* értelmezzük.

Ekvivalencia

Két rendszer esetén az állapotkombinációk részhalmazához állapotkombinációt rendelő $f: \bar{S} \rightarrow \bar{S}'$ függvény nem módosít az állapotkombinációk részhalmaz-viszonyain, ha $\forall s_1, s_2 \in \bar{S}$ esetén,

1. ha $s_1 = s_2$, akkor $\emptyset \subset f(s_1) = f(s_2) \subseteq 2^S$,
2. ha $s_1 \subset s_2$, akkor $\emptyset \subset f(s_1) \subset f(s_2) \subseteq 2^S$,
3. ha $s_2 \subset s_1$, akkor $\emptyset \subset f(s_2) \subset f(s_1) \subseteq 2^S$.

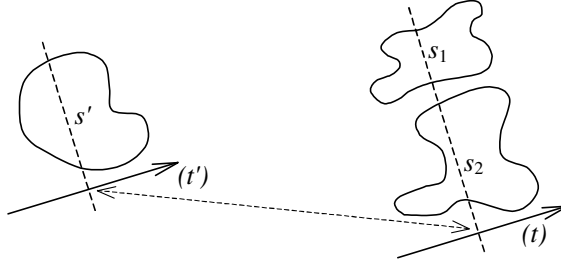
∇ Rendszerek ekvivalenciája, rendszer átjelölése. A $\sigma = (\mathcal{T}, S, P)$ *ekvivalens* a $\sigma' = (\mathcal{T}', S', P')$ rendszerrel (jelölés: $\sigma \equiv \sigma'$), vagy más szavakkal σ' a σ *átjelölése*, ha léteznek olyan

$\alpha_{\equiv_T}: T \rightarrow T'$ (szigorúan) monoton növekvő és

$\alpha_{\equiv_S}: \bar{S} \rightarrow \bar{S}'$ bijektív függvények,

hogy $\alpha_{\equiv_S}$ nem módosítja az állapotkombinációk részhalmaz-viszonyait, valamint $\forall t \in T$ esetén $P'(\alpha_{\equiv_T}(t)) = \alpha_{\equiv_S}(P(t))$.

Determinisztikus megadás esetén $\alpha_{\equiv_S}$ nem állapothalmazhoz rendel állapothalmazt, hanem állapotokat jelöl át.



3. ábra: Ekvivalencia: időpontok és tényleges állapotkombinációk átjelölése.

☞ *Rendszerek ekvivalenciája a bijekciók miatt a rendszerek ekvivalencia-relációja, azaz reflexív, szimmetrikus és tranzitív.*

§ Tetszőleges absztrakt rendszerek ekvivalensek.

Példa: Ekvivalencia.

Egy tó vizének hőmérsékletét egy rögzített időpontban és egy adott ponton mérjük meg és jegyezzük fel egy adott év minden napján. A hőmérsékleteket kerekítsük egész fokokra. A valóság vizsgált időpontjaihoz (pl. 1989 márc. 3, 13:00:00) a nap éven belüli sorszámát rendelve a rendszer egy ekvivalens átalakítását kapjuk. Ugyanakkor, pl. a 17 fok tekinthető a $16.5 < temp \leq 17.5$ értékek halmazának átjelöléseként.

▼ **Közvetlen átjelölés.** Az $\alpha_{\equiv} : \Sigma^{(\sigma)} \rightarrow \Sigma^{(\sigma')}$ közvetlenül hasonlítható rendszerek halmazából közvetlenül hasonlítható rendszerek halmazába képző, rögzített $\alpha_{\equiv_T}$ és $\alpha_{\equiv_S}$ függvényű ekvivalencia-transzformáció a közvetlen átjelölés.

„Közvetlen átjelölésen” közvetlenül hasonlítható rendszerek halmazából közvetlenül hasonlítható rendszerek halmazába történő adott leképezést nevezünk, míg a korábbi „rendszer átjelölése” elnevezés két rendszer közötti kapcsolatot ír le. A két szóhasználat így nem mond ellent, mivel egy közvetlen átjelölés alkalmazását követően a kiindulási rendszer átjelölését kapjuk.

Ugyanakkor egy rendszer átjelölésekor alkalmazott $\alpha_{\equiv_T}$ és $\alpha_{\equiv_S}$ függvények egy közvetlen átjelölést is meghatároznak.

Az ekvivalencia a későbbiekben az egyik központi fogalmunk lesz, mivel egy kiindulási rendszer átjelölése annak a tökéletes, más néven pontos, „veszteségmentes” ábrázolása.

§ Triviális ekvivalencia. Minden rendszer önmagával ekvivalens $\alpha_{\equiv_T}$ és $\alpha_{\equiv_S}$ identikus függvények esetén. Ezt *triviális ekvivalenciának* nevezzük.

Példa: Életjáték.

Egy számítógépes programmal megvalósított $n \times m$ -es életjáték esetén tekintsük azt a determinisztikus megadású rendszert, ahol az időpontok halmazát valós számokkal adjuk meg, a 0, mint indítási időponttal, de az átmenetek közötti időtartamokból csak egy-egy időpontban lesz értelmezett a rendszer. Állapothalmazként a memória megfelelő részletének állapotát vegyük. Rendszerünk átjelölhető a természetes számok időhalmazú és a $[0; 2^m - 1]$ közötti egészszámokkal megadható állapothalmazú rendszerré.

Életjáték egy konkrét lefutásának vízszintes vagy függőleges tükrözése, 90 fokra jobbra vagy balra forgatása (pl. a monitor oldalra fektetése) alapján kapott rendszer az eredeti átjelölése.

Az ekvivalencia megőrzi a változás változatosságát, amelyet a következő két állítás is szemléltet:

§ Konstans rendszerek ekvivalenciája. Két, azonos számosságú időn értelmezett konstans rendszer ekvivalens. Ekkor az egyik rendszer időpontjai értelmezhetők a másik rendszer időpontjainak átjelöléseként, illetve az egyik rendszer egyetlen, ténylegesen felvett állapotkombinációja értelmezhető a másik tényleges állapotkombinációjának átjelöléseként.

§ Ekvivalencia szigorúan változó rendszerek esetén. Két, azonos számosságú időn értelmezett szigorúan változó rendszer ekvivalens. Egy szigorúan változó (determinisztikus megadású) $\sigma = (T, T, S, P)$ rendszer ekvivalens a $\sigma' = (T, T, T, P')$, $P': t \mapsto t$ rendszerrel, azaz egy értelmezett időpont tekinthető az ott felvett állapot átjelölésének. Ha a ténylegesen felvett állapotokon definiálható olyan rendezés, melynél a változás szigorúan

monoton növekvő, akkor a felvett állapotok tekinthetők az időpontok átjelölésének.

Kompozíció és felbontás

▽ **Állapothalmazok egyszerű kompozíciója és kompozíciója.** Az $S_{l \in L}$ állapothalmazok $S = \otimes_{l \in L} S_l = \{l \mapsto s : l \in L \wedge s \in S_l\}$ (általánosított) direkt szorzatát az *állapothalmazok egyszerű kompozíciójának* nevezzük. Az L indexhalmaz elemeit a kompozíció részeihez, „helyeihez” rendelt azonosítóként, „megnevezéseként” értelmezzük. Az S_l a kompozíció l helyének elemeit határozza meg, amelyet $S^{(l)}$ (S kompozíció l része) vagy S_l (S kompozíció leszűkítése l részre) módon is jelölhetünk.

Az S az $S_{i \in I}$ állapothalmazok L indexhalmazon vett $\pi : I \rightarrow L$ (π szűrjektív) *kompozíciója*, ha $\forall i \in I$ esetén $S_i = S^{(\pi(i))}$. Tehát minden S_i meghatározza a kompozíció $\pi(i)$ részét, valamint, ha egy helyhez több halmazt is rendelünk, akkor azoknak definíció szerint meg kell egyezniük. A kompozíciót az $\otimes_{i \in I}^{(\pi : I \rightarrow L)} S_i^{(\pi(i))}$, vagy rövidítve a $\otimes_{i \in I}^\pi S_i$ módon jelöljük.

Állapothalmazok kompozíciójának elemei az *állapotkompozíciók*, amelyek a kompozíció helyein felvehető lehetséges állapotok sémáját töltik ki.

☞ *Állapothalmazok egyszerű kompozíciója az állapothalmazok kompozíciójának speciális esete, ekkor a π pozícionáló függvény identikus.*

Állapothalmazok egyszerű kompozíciója esetén az $s_i \in S$ összetett állapotokat az L indexhalmaz által megadott helyekre bontjuk fel. Az összetett állapot az adott helyen felvett állapotok direkt-szorzata lesz.

Állapothalmazok kompozíciója esetén megengedjük egy további pozícionáló-függvény létezését, amely meghatározza, hogy adott részállapot melyik helyet adja meg, azonban ha több részállapot is ugyanarra a helyre vonatkozik, akkor azoknak természetesen meg kell egyezniük.

▽ **Egyszerű kompozíció.** A $\sigma = (\mathcal{T}, S, P)$ rendszert a $\sigma_{l \in L} = (\mathcal{T}, S_l, P_l)$ rendszerek *egyszerű kompozíciójának* nevezzük, ha

$S = \otimes_{l \in L} S_l$ (egyszerű kompozíció), valamint minden $P_l(t)$ a σ rendszer $t \in \mathcal{T}$ -ben felvett összetett lehetséges állapotának l -hez tartozó komponense által felvehető állapotokat határozza meg: $\forall t \in \mathcal{T}$ esetén $P(t) = \otimes_{l \in L} P_l(t)$.

Példa: Egyszerű kompozíció.

Egy, a számjegyeit hét-hét vonallal ábrázoló digitális óra kijelzője, illetve annak aktív elemei megadhatók például az  óra és a perc kijelzőrészek egyszerű kompozíciójaként vagy a négy kijelzett számjegy egyszerű kompozíciójaként, vagy a számjegyenkénti 7 vonal 28 elemének egyszerű kompozíciójaként.

✓ Kompozíció. A $\sigma = (\mathcal{T}, S, P)$ rendszer a $\sigma_{i \in I} = (\mathcal{T}, S_i, P_i)$ rendszerek *kompozíciója* (jelölés: $\sigma = \otimes_{i \in I} \sigma_i$ vagy a pozicionálófüggvényt hangsúlyozva: $\sigma = \otimes_{i \in I}^{\pi} \sigma_i$), ha létezik a lehetséges helyek L indexhalmazába képző, a σ_i -nek a kompozícióban elfoglalt helyét meghatározó $\pi: I \rightarrow L$ szürjektív függvény, hogy $S = \otimes_{i \in I}^{(\pi: I \rightarrow L)} S_i^{(\pi(i))}$, valamint minden $P_i(t)$ a σ rendszer t -ben felvett összetett lehetséges állapotának meghatározza a σ_i -hez tartozó komponense által felvehető állapotokat, ha értelmezi a t időpillanatot:

$$P^{(l)}(t) = \begin{cases} P_i(t) & \text{ha } \exists i \in \pi^{-1}(l): t \in \mathcal{T} \\ S_l & \text{egyébként} \end{cases}$$

Ha egy rendszer azonos helyének azonos időpillanatban felvett állapotát több rendszer is értelmezi, akkor definíció szerint ezeknek az állapotoknak meg kell egyezniük.

☞ *Egyszerű kompozíció a kompozíció speciális esete.* Ekkor π identikus.

Példa: Életjáték.

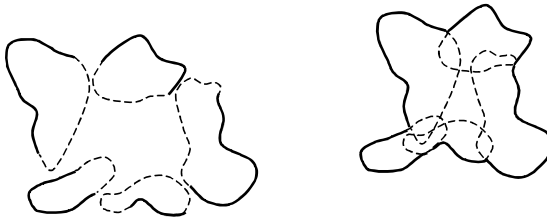
Egy $n \times m$ -es életjáték esetén az állapothalmaz megadható az $\{\text{üres}, \text{élő}\}$ kételemű halmaz nm -szeres Descartes-szorzataként, azaz egyszerű kompozíciójaként. Ekkor az életjáték felbontható az önálló sejtek kompozíciójára. A teljes állapot egyetlen számértékként történő ábrázolása ennek a formának az *átjelölése* lesz.

A sejtenként történő felírás mellett az életjátékot felbonthatjuk nagyobb egységekre is. Az egységek általános kompozíció esetén át is

fedhetik egymást, ekkor azonban a több részben is szereplő sejteknek azonos módon kell változni.

▼ **Komponens, rész, vagy alrendszer.** A σ rendszer a σ' része, más-képpen *komponense* vagy *alrendszere*, ha σ' előáll a σ és esetleges további rendszerek kompozíciójaként.

▼ **Dekompozíció vagy felbontás, kompozit.** Egy σ rendszer σ_i rendszerek kompozíciójaként történő felírását a rendszer *dekompozíciójának* vagy *felbontásának* nevezzük. A rendszert **kompozitnak** nevezzük, ha hangsúlyozni akarjuk, hogy azt kompozícióként tekintjük (minden rendszer tekinthető kompozícióként).



4. ábra. Egyszerű kompozíció és kompozíció

☞ *Egyszerű kompozíció és kompozíció esetén az eredmény értelmezett időpontjai* a kiindulási rendszerek értelmezett időpontjainak uniója lesz ($T = \bigcup_{i \in I} T_i$). Tehát csak azok az időpontok nem lesznek értelmezve, amelyek egyetlen kiindulási rendszer sem értelmez.

☞ *Az egyszerű kompozíció a kompozíció speciális esete.* Ha adott kompozícióban a pozíciónáló-függvényt identikusra cseréljük, akkor az eredeti rendszer átjelölését kapjuk, azaz az általános kompozíció nem jelent nagyobb változatosságot. Egy rendszer ábrázolásakor (lásd később) így elegendő csak egyszerű kompozíciót alkalmaznunk, az általános kompozíció nem feltétlenül szükséges, bár ekkor előfordulhat, hogy bizonyos részek változását feleslegesen több rész is meghatározza, azaz nagyobb lehet a redundancia. Az ábrázolás technikája szempontjából mindez azt jelenti, hogy egy „pontos ábrázoláshoz” elegendő felsorolnunk a részeket, de nem kell

megadnunk, hogy az ábrázolás mely része a rendszer mely komponensét írja le, amennyiben ez a pozícionálás nem változik.

§ Kompozíció és egyszerű kompozíció eredményei ekvivalensek. Adott $\sigma_{i \in I}$ rendszerek rögzített π függvényű kompozíciója és egyszerű kompozíciója – ha mindkét átalakítás végrehajtható – ekvivalens rendszereket eredményeznek.

§ Ekvivalens átalakítás és redundancia. Ha a rendszer állapotának egy vagy több komponensét megismételjük, akkor egy átjelölést, az eredetivel ekvivalens rendszert kapunk. Ellentétes irányban: a rendszer azonosan változó komponensei egyetlen komponenssel helyettesíthetők, így a redundancia eltüntethető.

Egyszerű kompozíció eredményének bármely része egy adott időpillanatban komplex állapotának bizonyos komponenseivel utalhat más részek állapotaira. Az azonos időn értelmezett rendszerekből egyszerű állapotkompozícióval létrehozott rendszer még redundánsként megismételheti bizonyos állapotait. Egy ekvivalencia segítségével ez a redundancia eltüntethető.

Példa: Egyszerű kompozíció és kompozíció.

Egy digitális óra számjegyei előállíthatók a vízszintes középső vonalak és a fölötte, valamint a vízszintes középső és az alatta elhelyezkedő vonalak – mint két komponens – kompozíciójaként. A két komponens egyszerű kompozíciója esetén redundánsan megismétlődnének az azonos változású középső vonalak. Az így kapott rendszer ekvivalens az eredeti kompozícióval.

✓ Azonos idejű és időre vonatkozó kompozíció és részekre bontás.

Az *azonos idejű kompozíciót* a T_i halmazok azonosságával definiáljuk. Ellenkező esetben *időre vonatkozó kompozícióról* beszélünk. Ekkor bizonyos σ_i rendszerek különböző T_i időpillanatokat is értelmeznek, így azok a rendszer változásának különböző időpontjait, a σ más-más időpontokban felvett állapotait írják le. Hasonló módon határozható meg az *azonos idejű és az időre vonatkozó részekre bontás*.

Példa: Időre vonatkozó szétbontás.

A rendszer állapotait az időre vonatkozóan kettébontjuk, így $\pi(1)=\pi(2)$.

t	S
1	1
2	4
3	2
4	3
5	2

t	s_1	s_2
1	1	
2	4	
3	2	2
4		3
5	2	

Az üresen hagyott helyekhez tartozó időpillanatokban a felbontás részei nincsenek értelmezve, azaz meghatározatlan állapotúak. Azokban az időpontokban, ahol mindkét rendszer értelmezve van, az állapotoknak meg kell egyezniük.

Egy rendszer kompozícióként történő vizsgálata során már nem csak azt figyelhetjük meg, hogy milyen időpontokat értelmez a rendszer, hanem azt is, hogy adott rész adott időpontban felvett állapota értelmezett („ismert”), vagy tetszőleges. Korábbi állításunk alapján a rendszer egy értelmezett időpontja esetén még bizonyos (de nem az összes) részeinek állapotai tetszőlegesek is lehetnek.

Ez a szempont a kompozitok egy új, a rendszerek normalizált megadásával analóg megadási formához vezet el bennünket, ahol az állapotfüggvény külön-külön határozza meg adott komponens adott időpontban felvett állapotát, mely értelmezési tartományából kiemeljük az „értelmezetlen” idő-hely elemeket.

✓ Kompozícióként megadott rendszer. Egy rendszer (*normalizált*) kompozícióként történő $\sigma = (\mathcal{T}, L, \overline{\mathcal{T} \times L}, S, P)$ megadása esetén $\mathcal{T}$ a rendszer változásának ideje, L a lehetséges helyek halmaza, adott S_l ($l \in L$) halmazok egyszerű kompozíciójaként előálló S a rendszer lehetséges állapotainak halmaza, a változás $P: \overline{\mathcal{T} \times L} \rightarrow \bigcup_{l \in L} (2^{S_l} \setminus \{\emptyset, S_l\})$, $P(t, l) \in 2^{S_l} \setminus \{\emptyset, S_l\}$ függvénye pedig az adott időpontban a kompozíció adott része által felvehető állapotokat határozza meg. A $(t, l) \in \mathcal{T} \times L \setminus \overline{\mathcal{T} \times L}$ elempárok esetén a t időpillanatban a kompozíció l része tetszőleges, értelmezetlen állapotú. *Determinisztikus kompozícióként történő megadás* esetén a változás függvénye $P: \overline{\mathcal{T} \times L} \rightarrow \bigcup_{l \in L} S_l$ és $P(t, l) \in S_l$ alakú.

☞ Egy kompozícióként megadott rendszernek a már korábban ismertett megadási módokba és vissza történő alakítása egyértelmű transzformáció.

Kompozícióként megadott rendszer esetén kiemeljük az adott időpontokban értelmezetlen állapotú részeket. A $\mathcal{T}$ halmaz újabb, értelmezetlen időpontokkal történő bővítéséhez hasonlóan egy kompozit újabb, értelmezetlen állapotú részekkel történő kiegészítése – bár más rendszert eredményez – lényegében nem módosít a változáson.

✓ Kompozit egyszerűsített megadása. Kompozit $\sigma = (\overline{\mathcal{T} \times L}, S, P)$ elemhármassal történő egyszerűsített megadása a különböző $\mathcal{T}_i$ időbe és L_i lehetséges helyek halmazába ágyazott $(\mathcal{T}_i, L_i, \overline{\mathcal{T} \times L}, S, P)$ rendszerek egy (tetszőleges) reprezentációs-rendszerét jelöli.

Kompozit egyszerűsített megadása a részekre bontott változás lényegét emeli ki, mivel eltekint az adott időpontokban és helyeken határozatlan állapot-kombinációktól. A változás „lényegében” ugyanaz marad, ha ahhoz értelmezetlen időpontokat vagy helyeket veszünk hozzá, vagy hagyunk el abból.

✓ Azonosnak tekintett rendszerek. A $\sigma = (\mathcal{T}, L, \overline{\mathcal{T} \times L}, S, P)$ és $\sigma' = (\mathcal{T}', L', \overline{\mathcal{T}' \times L'}, S', P')$ azonosnak tekintett rendszerek (jelölés: $\sigma \Leftrightarrow \sigma'$, azaz a két rendszer azonosnak tekintett, ill. $\sigma' = \sigma^{\Leftrightarrow}$, azaz a σ -val azonosnak tekintett $\sigma^{\Leftrightarrow}$ rendszer megegyezik σ' -vel), ha $\overline{\mathcal{T} \times L} = \overline{\mathcal{T}' \times L'}$, $\overline{S} = \overline{S'}$ és $\forall (t, l) \in \overline{\mathcal{T} \times L}$ esetén $P(t, l) = P'(t, l)$. Egyetlen komponensből álló rendszer esetén a feltétel redukálható: $\mathcal{T} = \mathcal{T}'$, $\overline{S} = \overline{S'}$ és $\forall t \in \mathcal{T}$ esetén $P(t) = P'(t)$.

☞ $A \Leftrightarrow$ ekvivalencia-reláció.

☞ Az azonos egyszerűsített megadású kompozitok $(\sigma_{i \in I} = (\overline{\mathcal{T} \times L}, S, P))$ azonosnak tekintett rendszerek.

Ha bizonyos rendszerek azonosnak tekinthetők, akkor azok lényegében azonos módon változnak, mely viszonyt a következő megjegyzések is szemléltetik:

☞ A tetszőleges absztrakt rendszerek azonosnak tekinthetők.

☞ Rendszer önmagával azonosnak tekinthető.

☞ *Az azonosnak tekintett rendszerek ekvivalensek.*

✓ **Triviális kompozíció és triviális rész.** Minden rendszer önmaga kompozíciója és egyben önmaga része. A kompozíciónak az önmagával azonosnak tekintett (speciális esetben azonos) rendszert létrehozó változatát **triviális kompozíciónak**, részekre bontás esetén **triviális résznek** nevezzük.

☞ *A kompozíció, illetve a részképzés – mint két rendszer közötti reláció – tranzitív és aszimmetrikus.* Rendszer részének a része egyben a kiindulási rendszer része is (a kompozíciók zárójelezése felbontható), illetve kompozit nem lehet a része egyetlen nem triviális komponensének sem.

Feltétel, következmény, függetlenség és ellentmondás

Az azonosnak tekintett rendszerek lényegében ugyanazt a változást határozzák meg, így ez a definíció lehetővé teszi, hogy a közvetlen hasonlíthatóság fogalmát kiterjesszük. A rendszerek közvetlenül hasonlíthatók a $\mathcal{T}$ és S halmazaik azonossága esetén. A hasonlíthatóság általánosabb definíciója esetén megengedjük, hogy az eredeti rendszerekkel egy-egy azonosnak tekintett rendszerrel végezzük el a vizsgálatot, melyeknek már közös értelmezett idő- és állapothalmazúaknak kell lenni. A kiindulási rendszerrel azonosnak tekintett, de a közvetlen hasonlíthatóság érdekében módosított idő- és állapothalmazú rendszert az eredeti rendszer hasonlíthatóvá történő átjelölésének nevezzük.

✓ **Hasonlíthatóvá átjelölés, (közvetve) hasonlítható rendszerek.** A σ a σ' -nek egy adott σ'' -vel *hasonlíthatóvá* való átjelölése, ha az a σ' -vel azonosnak tekintett és σ'' -vel közvetlenül hasonlítható ($\sigma' \leftrightarrow \in \Sigma(\sigma'')$). A $\sigma_{i \in I}$ rendszerek (közvetve) *hasonlíthatók* (jelölés: $\Sigma(\sigma_i)^*$), ha minden σ_i -hez létezik vele azonosnak tekintett $\sigma_i^{\leftrightarrow}$ rendszer, amelyek közvetlenül hasonlíthatók. A $\Sigma(\sigma)^*$ jelöli a σ -val (közvetve) hasonlítható rendszerek halmazát.

A hasonlíthatóvá átjelölés adott rendszer esetén bizonyos időpontokban értelmetlen állapotú helyek, speciális esetben egyáltalán nem értelmezett időpontok, illetve helyek elhagyását vagy hozzáadását jelenti.

☞ *Kompozíció és részei (közvetve) hasonlíthatók.*

A hasonlíthatóság lehetővé teszi, hogy a kiindulási $\sigma_{i \in I}$ rendszerek hasonlíthatóvá átjelölt $\sigma_{i \in I} \Leftrightarrow (\mathcal{T}, S, P_i)$ rendszereinek a közös $t \in \mathcal{T}$ időpontokban felvett állapotkombinációit a halmazműveletek segítségével összehasonlítsuk.

∇ Ellentmondó rendszerek. A $\sigma_{i \in I} \Leftrightarrow (\mathcal{T}, S, P_i)$ hasonlítható rendszerek *ellentmondók*, ha létezik olyan $t \in \mathcal{T}$ időpont, hogy $\bigcap_{i \in I} P_i(t) = \emptyset$.

Ellentmondó rendszerek esetén létezik legalább egy olyan időpont, amelyre a rendszerek diszjunkt állapotkombinációkat határoznak meg (teljesen mást állít az egyik, mint a másik).

Közvetlenül hasonlítható rendszerek esetén az időpontokban felvett állapotkombinációkkal halmazműveletek is végezhetők, melyekkel a rendszerek közötti műveleteket is definiálhatunk.

∇ Rendszerek metszete. A $\sigma = (\mathcal{T}, S, P)$ rendszert a közvetlenül hasonlítható és nem ellentmondó $\sigma_{i \in I} = (\mathcal{T}, S, P_i)$ *rendszerek metszetének* nevezzük (jelölés: $\sigma = \bigcap_{i \in I} \sigma_i$), ha $\forall t \in \mathcal{T}$ -re $P(t) = \bigcap_{i \in I} P_i(t)$. Mivel a rendszerek nem ellentmondók, ezért a metszet soha nem lehet üreshalmaz.

☞ *Egy adott rendszerrel hasonlítható, de nem ellentmondó rendszerek halmazán a metszet egy művelet.*

☞ *Rendszerek metszete esetén $T \subseteq \bigcup_{i \in I} T_i$, azaz az eredményül kapott rendszer bármely kiindulási rendszer értelmezett időpontját értelmezi.*

∇ Rendszerek uniója. A $\sigma = (\mathcal{T}, S, P)$ rendszer a közvetlenül hasonlítható $\sigma_{i \in I} = (\mathcal{T}, S, P_i)$ *rendszerek uniója* (jelölés: $\sigma = \bigcup_{i \in I} \sigma_i$), ha $\forall t \in \mathcal{T}$ -re $P(t) = \bigcup_{i \in I} P_i(t)$.

☞ *Az unió egy művelet a közvetlenül hasonlítható rendszerek halmazán.*

☞ *Rendszerek uniója esetén* $T \subseteq \bigcap_{i \in I} T_i$, azaz az eredményül kapott rendszer legfeljebb a közösen értelmezett időpontokat értelmezi. Közülük is csak azokat, amelyek esetén a rendszerek állapot-lehetőségei együttesen nem adják meg a teljes állapothalmazt.

Példa: Rendszerek uniója.

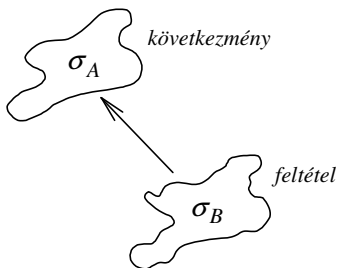
Egy korábbi példánkban szerepelt egy digitális óra, mely véletlenszerűen váltott a 12/24 órás kijelzés között. Ha eltekintünk az éppen aktuális kijelzési módtól, akkor a két lehetséges változat unióját kapjuk.

A közvetlenül hasonlítható rendszereken nem csak az unió és metszet műveleteit végezhetjük el, hanem az időpontokban felvett állapotkombinációk vizsgálatával maguk a rendszerek is összehasonlíthatók, mely viszonyok a (közvetve) hasonlítható rendszerekre is kiterjeszthetők.

Az összehasonlítást az ellentmondó viszonytal analóg módon úgy lehet elképzelni, mintha a rendszerek valamely közös változásra vonatkoznának. A rendszerek így leírhatják lényegileg ugyanazt a változást, ha azok azonosnak tekinthetők. A rendszerek lehetnek ellentmondók, ha egy vagy több időpontban gyökeresen ellentmondó (értsd: diszjunkt) állapotkombinációkat állítanak. Az is lehetséges, hogy egy rendszer minden időpontjában egy másik rendszerrel azonos, vagy annál bővebb állapotkombinációkat engedélyez, azaz kevésbé pontosan írja le a változást.

Az összehasonlítás alapja az adott időpontban felvehető állapotlehetőségek viszonyának vizsgálata, mely kiterjeszthető a rendszerek által értelmezett összes időpontra. Rendszerek összehasonlítása közvetlenül hasonlítható rendszerek esetén egyszerűen az időpontokban felvett állapotkombinációk összehasonlításával végezhető el. A hasonlíthatóvá átjelölés segítségével az eszközkészlet a (közvetve) hasonlítható rendszerek viszonyára is kiterjeszthető.

▽ **Következmény és feltétel vagy példa.** A σ_A rendszer a (közvetve) hasonlítható σ_B rendszer *következménye*, illetve σ_B a σ_A *feltétele* vagy *példája*, ha σ_A -nak a σ_B -vel történő, $\sigma_A^{\leftrightarrow} = (T, S, P_A)$ hasonlíthatóvá átjelölése esetén $\forall t \in T$ -re $P_A(t) \supseteq P_B(t)$. Jelölés: $\sigma_A \Leftarrow \sigma_B$, illetve $\sigma_B \Rightarrow \sigma_A$, közvetlenül hasonlítható rendszerek esetén $\sigma_A \supseteq \sigma_B$, illetve $\sigma_B \subseteq \sigma_A$, valamint valódi tartalmazás esetén $\sigma_A \supset \sigma_B$, illetve $\sigma_B \subset \sigma_A$.



5. ábra. Következmény és feltétel

☞ „Feltétel” és „következmény” tranzitív, reflexív és antiszimmetrikus relációk.

A következmény az egyes időpontokban felvehető állapotok halmazának bővítési lehetősége – amely speciális esetben (ha az eredmény a teljes állapothalmaz) az *időpontoktól történő eltekintést*, annak nem értelmezését is jelentheti. Ezzel a kiindulási rendszer meghatározottságát csökkentettük, az eltekintett időpontok szélső esetében egészen a meghatározatlan állapotig.

A feltétel és a következmény alkalmas a lényegében részletezettebb, ill. a lényegében vázlatosabb változás megfogalmazására:

☞ Minden rendszer önmaga feltétele és következménye.

☞ Minden rendszer a feltételének (példájának) a következménye.

☞ Minden rendszer a vele hasonlítható tetszőleges absztrakt rendszer feltétele (példája), ill. hasonlítható rendszereknek a velük hasonlítható tetszőleges absztrakt rendszer a következménye.

§ **Azonosnak tekintett rendszerek – feltétel és következmény.** A hasonlítható σ_A és σ_B esetén $\sigma_A \Leftarrow \sigma_B$ és $\sigma_A \Rightarrow \sigma_B$ pontosan akkor teljesül, ha a két rendszert azonosnak tekintjük ($\sigma_A \Leftrightarrow \sigma_B$).

▽ **Többszörös feltétel és többszörös következmény.** Ha egy σ_A rendszer adott $\sigma_{i \in I}$ rendszerek következménye, akkor *többszörös feltételről*

(jelölés: $(\sigma_i) \Rightarrow \sigma_A$), ha egy σ_B rendszer adott $\sigma_{j \in J}$ rendszerek feltétele, akkor *többszörös következményről* (jelölés: $\sigma_B \Rightarrow (\sigma_j)$) beszélünk.

▽ Hasonlítható absztrakció, hasonlítható pontosítás. A σ_A rendszer a vele hasonlítható $\sigma_{i \in I}$ rendszerek *hasonlítható absztrakciója*, ha minden σ_i -nek a σ_A a következménye. A σ_B rendszer a vele hasonlítható $\sigma_{i \in I}$ rendszerek *hasonlítható pontosítása*, ha minden σ_i -nek a σ_B a feltétele.

☞ *Egyetlen kiindulási rendszer esetén a hasonlítható absztrakció és a következmény, illetve a hasonlítható pontosítás és a feltétel fogalmi ekvivalens fogalmak.*

☞ *A $(\sigma_i) \Rightarrow \sigma_A$ többszörös feltétel esetén a σ_A a σ_i rendszerek hasonlítható absztrakciója; a $\sigma_B \Rightarrow (\sigma_j)$ többszörös feltétel esetén a σ_B a σ_j rendszerek hasonlítható pontosítása.*

§ Következmény megadható unióként esetleges kiegészítő rendszerek (elegendő egy) felvétele segítségével. Adott kiindulási rendszerek hasonlíthatóvá történő átjelölését követően végezzük el az unió műveletet. A művelet eredményének a következménye lesz bármely olyan rendszer, amelynek a többszörös feltételei a kiindulási rendszerek. Ezt a következményt megkaphatjuk úgy is, ha a kiindulási rendszerekhez felveszünk további (elegendő legfeljebb egy) rendszereket, amelyeknek adott időponthoz tartozó uniói megadják a hiányzó $P(t) \setminus \bigcup_{i \in I} P_i(t)$, eltekintett időpontok esetén pedig az $S \setminus \bigcup_{i \in I} P_i(t)$ állapotokat. Rendszerek együttes következményének hasonlítható absztrakciója tehát további kiindulási rendszerek implicit felvételeként is értelmezhető.

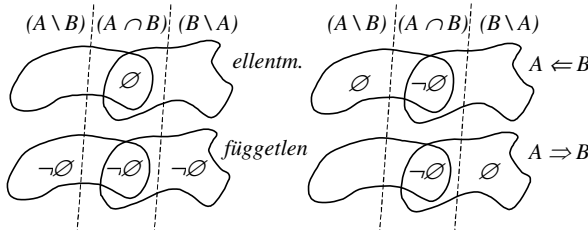
☞ *Kompozit a részeinek a feltétele (példája).*

§ Kompozit a részeinek a hasonlítható pontosítása; kompozitnak a részei a többszörös következményei.

▽ Függés és függetlenség. A hasonlítható σ_A és σ_B rendszerek *függők* (jelölés: $\sigma_A \wr \sigma_B$), ha σ_A a σ_B rendszer feltétele vagy következménye.

A hasonlítható σ_A és σ_B rendszerek *függetlenek*, ha nem függők és nem ellentmondók.

✓ **Rendszerek összehasonlítása.** Két hasonlítható rendszer *összehasonlítása* a rendszerek feltétel-következmény (azaz függő), független vagy ellentmondó viszonyának a megállapítása.



6. ábra. Rendszerek összehasonlítása

✓ **Rendszerek konjunkciója vagy magrendszere.** A $\sigma = (T, S, P)$ rendszer a kiindulási $\sigma_{i \in I}$ rendszerek konjunkciója (jelölés: $\sigma = \bigwedge_{i \in I} \sigma_i$), ha léteznek a σ_i -vel azonosnak tekintett $\sigma_i^{\Leftrightarrow} = (T, S, P_i)$ nem ellentmondó rendszerek, melyek metszete σ . Rendszerek adott halmazának konjunkciói azonosnak tekintett rendszerek, így az eredményrendszer T és S halmazának rögzítése mellett a nem ellentmondó rendszerek konjunkciója egyértelmű transzformáció.

☞ *Konjunkció eredménye a kiindulási rendszerek hasonlítható pontosítása; a konjunkciónak a kiindulási rendszerek a többszörös következményei.*

§ **Függő és független rendszerek magrendszere.** Két függő (hasonlítható) rendszer konjunkciója megegyezik a két rendszer közül a feltétellel. Független rendszerek konjunkciója a kiindulási rendszerek feltétele. A kiindulási rendszerek a magrendszer többszörös következményei.

Példa: Hasonlítható rendszerek viszonya.

A táblázatban soronként két *helyzet* egyetlen, azonos időpillanatban van értelmezve és közös állapothalmazuk a valós számok halmaza. A táblázat az időpontban a párok felvehető állapotkombinációit, viszonyukat és metszetüket adja meg.

σ_A	<i>viszony</i>	σ_B	$\sigma_A \wedge \sigma_B$
[0;1]	$\Rightarrow$	[0;2]	σ_A
[0;3]	$\Leftarrow$	[0;2]	σ_B
[0;2]	függetlenek	[1;3]	[1;2]
[0;1]	ellentmondók	[2;3]	nincs értelmezve

Korlátozó és példány

✓ **Korlátozó megszorítás vagy korlátozó.** A kiindulási rendszer *korlátozó megszorításának*, vagy röviden a *korlátozójának* nevezzük azt a következményét, amely nem (a hasonlítható) tetszőleges absztrakt rendszer.

Példa: Következmény, korlátozó.

Mérés hiányában egy tő hőmérsékletére vonatkozó kérdésre csak az összes lehetséges hőmérsékleti értéket jelentő „tetszőleges”, „bizonytalan” választ adhatjuk. Ezzel a tényleges rendszer következményét kapjuk. Egy, erre az időpontra, azaz erre a *situációra* korlátozott rendszer esetén ez a következmény egy tetszőleges absztrakt rendszert eredményez, ezért ez nem a *situáció korlátozója*. Ha az érték tetszőlegességét egy tartományra, például a $[-100^\circ; 100^\circ]$ fokra szűkítjük, akkor már egy korlátozót kapunk. Maga a mérés szintén korlátozóként jelenik meg, mivel a hibahatáron belüli tartományra szűkíti a hőmérsékletet. Szintén korlátozó a pontos hőmérséklet, mivel minden rendszer egyben önmaga következménye is.

Tehát a tényleges hőmérséklet korlátozója önmaga, illetve a mérés által kapott tartomány, annak korlátozója a $[-100^\circ; 100^\circ]$ tartomány, melynek következménye a hasonlítható tetszőleges absztrakt rendszer.

Vegyünk egy, a mérések által kapott rendszert. Ha egyetlen mérés kivételével az összes többi állapotát a tetszőlegességgel helyettesítjük, azaz kiragadunk egyetlen mérést, akkor ezzel a rendszer egy korlátozóját kapjuk, amely egyben a valóság korlátozója is lesz. Tehát már egyetlen megfigyelés is korlátozza a rendszert, azaz kiemeli a tetszőlegességből.

A korlátozó megszorítás korlátozza, azaz a tetszőlegességből kiemeli a rendszert, azt valamilyen pontossággal meghatározza. Általános jellegénél fogva éppen ezért alkalmas a rendszerre vonatkozó információ definíciójára, így a későbbiekben ez az egyik központi fogalmunk lesz.

✓ **Rendszer ellentéte vagy negáltja.** Egy $\sigma = (T, T, S, P)$ rendszer ellentétén azt a $\sigma' = (T', T, S, P')$ rendszert értjük, mely minden $t \in T$ időpillanatban $P'(t) = S \setminus P(t)$, azaz „nem $P(t)$ ” állapotban van.

☞ Egy tetszőleges absztrakt rendszer ellentéte önmaga.

☞ Rendszernek létezik vele közvetlenül hasonlítható ellentéte.

☞ Rendszer és közvetlenül hasonlítható ellentétének uniója a hasonlítható tetszőleges absztrakt rendszer.

☞ Rendszer és ellentéte hasonlítható és pontosan akkor ellentmondó rendszerek, ha a kiindulási rendszer nem egy tetszőleges absztrakt rendszer.

✓ **Korlátozás.** Rendszer példáját korlátozásnak nevezzük, ha csak a már eddig is értelmezett pozíciók lehetséges állapotkombinációit szűkíti tovább.

✓ **Példány.** A σ rendszer példány, ha nem adható meg olyan korlátozás, amely a σ rendszerrel nem azonosnak tekintett (hanem pontosabb) rendszert eredményez. A példány-jelleget a $\overline{\sigma}$ jelöléssel hangsúlyozhatjuk.

☞ Példány esetén adott időpont-hely párokon az értelmezett állapotkombinációk egyelemű halmazok.

§ **Egy rendszer nem azonosnak tekintett példányai ellentmondó rendszerek.**

✓ **Következmény példánya, példányképzés vagy konkretizálás.** Rendszer egy σ következményének $\overline{\sigma}$ korlátozása a σ példánya, ha nem korlátozható a σ pontosabb korlátozójává. Rendszer következményének példányt létrehozó korlátozását példány képzésének vagy konkretizálásnak nevezzük.

§ **Példány következményének a példánya a rendszer komponense.**

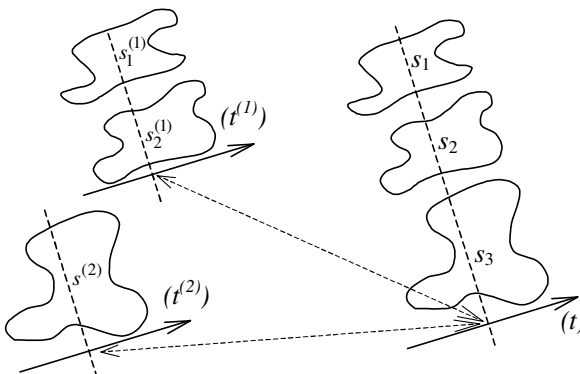
Absztrakció és pontosítás

Azonosnak tekintett rendszereknek neveztük azokat a rendszereket, amelyek lényegében ugyanazt a változást határozzák meg, bár más teljes idő- illetve állapothalmaz keretébe helyezhetik azt. A feltétel és következmény-fogalmunk segítségével a hasonlítható rendszerek esetén meg tudjuk fogalmazni a lényegében részletezettebben, ill. vázlatosabban leírt változást.

Az ekvivalencia-fogalom alkalmas arra, hogy kifejezze adott rendszerek-nél a változás azonos változatosságát, amely általában már nem a lényegében azonos változást jelenti, de adott rendszer bijektív módon („információ-vesztés” nélkül) a vele ekvivalens rendszerré átjelölhető.

Az azonosnak tekintett rendszerek a lényegében azonos változást, az ekvivalens rendszerek pedig az azonos változatosságú változás viszonyát írják le. A változás lényegének részletezettebb/vázlatosabb fogalmának, azaz a feltétel és a következmény mintájára a nagyobb, illetve kisebb változatosságú változások viszonya is definiálható, amit pontosításnak, illetve absztrakciónak fogunk nevezni.

✓ **Nemdeterminisztikus absztrakció.** A σ rendszer a $\sigma_{i \in I}$ rendszerek *nemdeterminisztikus absztrakciója*, ha léteznek a σ_i rendszerekkel ekvivalens olyan közvetlenül hasonlítható σ'_i rendszerek, amelyek uniója a σ rendszer.



7. ábra. Nemdeterminisztikus absztrakció a lefutási irányok egyesítése.

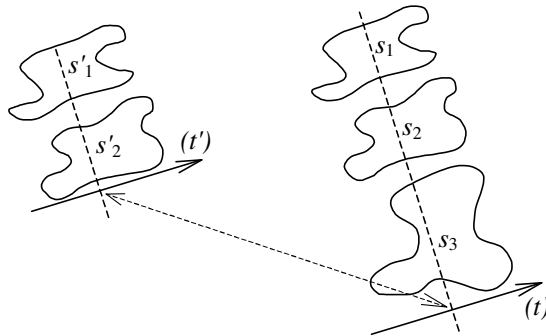
Nemdeterminisztikus absztrakció végrehajtásakor a kiindulási rendszerek közös időre és állapothalmazra történő (esetleges) átjelölése után több lehetséges lefutási irányt egyesítünk.

☞ *Az unió a nemdeterminisztikus absztrakció speciális esete: ekkor a kezdő átjelölések identikusak.*

Példa: Nemdeterminisztikus absztrakció.

Korábbi példánk alapján a tó hőmérsékletének mérését ismétljük meg tíz éven keresztül. A tíz rendszer uniójaként kapott rendszer segítségével – amely ez év egy adott napjához a hőmérsékletek legfeljebb tízelemű halmazát rendeli – megválaszolható például a következő kérdés: az elmúlt tíz évben február elsején a mérések időpontjaiban milyen hőmérsékletű volt a víz? A művelet kiindulási rendszereit egy-egy ekvivalens átalakítással kaptuk, amely például az 1989 febr. 1, 13:00:00 időponthoz a 32 értéket (mint az év 32-ik napja), a $16.5^\circ < temp \leq 17.5^\circ$ hőmérsékleti tartományhoz a 17 értéket rendeli.

✓ **Absztrakció.** A $\sigma = (T, S, P)$ rendszer a $\sigma_{i \in I} = (T_i, S_i, P_i)$ rendszerek *absztrakciója* vagy *általánosítása*, ha léteznek a σ -val ekvivalens $\sigma' = (T', S', P')$ és σ_i rendszerekkel ekvivalens, σ' -el közvetlenül hasonlítható $\sigma'_i = (T', S', P'_i)$ rendszerek, melyek hasonlítható absztrakciója (feltétele) σ' .



8. ábra. Absztrakció során egy adott időpont lehetséges állapotai bővíülhetnek.

Az absztrakció a kiindulási rendszerek esetleges átjelölése mellett az egyes időpontokban felvehető állapotok bővebb halmazba sorolásának lehe-

tősege, amely az *időpontoktól történő eltekintést*, annak nem értelmezését is jelentheti, illetve a kapott rendszer átjelölése.

§ **Nemdeterminisztikus absztrakció az absztrakció speciális esete.**

Nemdeterminisztikus absztrakció esetén a rendszerek kezdő átjelölése után az adott időpontban felvehető állapotok halmazának bővítése csak a kiindulási rendszerek időponthoz tartozó halmazainak uniója lehet. Időponttól csak abban az esetben tekinthetünk el, ha az unió a teljes állapothalmazt megadja. Nincs lehetőség az eredmény záró átjelölésére sem.

§ **Absztrakcióval ekvivalens rendszer megadható nemdeterminisztikus absztrakcióval**, mivel korábbi állításunk értelmében rendszerek következménye megadható unióként, esetleges kiegészítő rendszerek felvételével.

§ **Rendszer absztrakciója megadható unióval és ekvivalens átalakításokkal.** Az absztrakció eredményével ekvivalens rendszert adjuk meg nemdeterminisztikus absztrakcióként – egy alkalmas kiegészítő rendszer esetleges felvételével. Így az absztrakció átírható a kiindulási és az esetleges kiegészítő rendszer átjelölésének, az unió és egy záró átjelölésnek a formájára.

§ **Rendszer absztrakciója megadható következménnyel és átjelöléssel.** Egyetlen rendszer absztrakciója során végrehajtott kezdő átjelölés, hasonlítható absztrakció, majd záró átjelölés műveletei átcsoportosíthatók egy következményt (időpontokhoz tartozó állapotok halmazainak bővítését, azaz hasonlítható absztrakciót) követő átjelöléssé, vagy egy átjelölést követő hasonlítható absztrakcióvá.

✓ **Pontosítás vagy specializáció.** A σ' rendszer a σ pontosítása, specializációja, derivátuma vagy származtatása, ha σ a σ' absztrakciója.

(A derivátum-jelleget a $\bar{\sigma}$ jelöléssel hangsúlyozhatjuk.)

☞ *Feltétel és következmény – pontosítás és absztrakció.* Rendszer következménye a rendszer absztrakciója, illetve rendszer feltétele a rendszer pontosítása.

✓ **Többszörös pontosítás.** A σ a σ_i rendszerek többszörös pontosítása, ha minden i esetén σ_i a σ absztrakciója, azaz léteznek olyan absztrakciók, melyek σ -ra alkalmazva σ_i -ket eredményezik.

A pontosítás és az absztrakció alkalmas a változatosabb, ill. a kevésbé változatos változás viszonyának megfogalmazására. A pontosítás és az absztrakció ezért rendkívül erős eszközök, amit a következő megjegyzés és állítás is szemléltet:

☞ *Minden rendszer egy (bármely) tetszőleges absztrakt rendszer pontosítása.*

§ **Bármely rendszerből egy absztrakciót követő pontosítással tetszőleges rendszer képezhető.**

§ **Rendszerek ekvivalenciája és az absztrakció.** A σ rendszer pontosan akkor ekvivalens σ' -vel ($\sigma \equiv \sigma'$), ha mindkét rendszer egymás általánosítása és pontosítása is egyszerre.

✓ **Triviális absztrakció és pontosítás, triviális feltétel és következmény.** A kiindulási rendszerrel ekvivalens rendszert létrehozó általánosítást *triviális absztrakciónak*, pontosítás esetén *triviális pontosításnak* nevezzük. Rendszerrel vele azonosnak tekintett rendszert létrehozó következményt *triviális következménynek*, feltételt *triviális feltételnek* nevezzük. Ellenkező esetben absztraktabb, illetve pontosabb rendszerről beszélünk.

☞ *Az általánosítást és pontosítást két rendszer közötti viszonyként, relációként értelmezve igaz a tranzitivitás tulajdonsága, azaz ha egy általánosítás eredményén újabb általánosítást hajtunk végre, akkor a kapott rendszer egyben az eredeti absztrakciója is lesz. Ugyanez érvényes a pontosításra is.*

Típus

A típus fogalmát az értelmezett állapotok osztályokba csoportosításával, azok halmazának diszjunkt halmazokra való bontásával kapcsoljuk össze. Az egész számok halmazát felbonthatjuk például a 0 vagy páros, ill. a páratlan számok halmazaira, vagy egy más csoportosítás esetén a negatív, a 0 számot tartalmazó egyelemű halmaz és a pozitív számok halmazaira.

Egy, a halmazon értelmezett függvény a kiindulási halmazon egy osztályozást indukál. Invertálható függvény esetén minden állapot külön, egyelemű osztályban van. Ha azonban a függvény különböző értékekhez ugyanazt

az értéket is rendelheti, akkor az érték ősképei (az azonos értékű kiindulási elemek) többelemű csoportot alkotnak.

▼ **Tényleges vagy értelmezett állapotok.** A $\sigma = (\mathcal{T}, S, P)$ rendszer tényleges állapotai (nem állapotkombinációi) az S azon elemei, amelyek szerepelnek a ténylegesen felvett állapotkombinációkban. A tényleges vagy értelmezett állapotok jelölése: $\bar{S}$, így $\bar{S} = \bigcup_{x \in \bar{S}} x$. Nemdeterminisztikus esetben úgy tekintjük, hogy a tényleges állapotok és a tényleges állapotkombinációk halmaza megegyezik.

☞ *Állapotokon és állapotkombinációkon értelmezett függvény.* Két rendszer tényleges állapotainak halmazai között megadott $\alpha_{\sqrt{S}} : \bar{S} \rightarrow \bar{S}'$ függvény esetén az értelmezett állapotkombinációk között az közvetve megad egy $\alpha_{\sqrt{S}}^* : \bar{S} \rightarrow \bar{S}'$ leképezést, amely esetén $\alpha_{\sqrt{S}}^* : s \mapsto \bigcup_{\forall x \in s} \{\alpha_{\sqrt{S}}(x)\}$, azaz minden értelmezett állapotkombináció állapotait az $\alpha_{\sqrt{S}}$ -nek megfelelő állapokra cseréli. Mivel az $\alpha_{\sqrt{S}}$ nem feltétlenül bijektív, ezért az különböző kiindulási állapotokhoz ugyanazt az eredményállapotot is rendelheti, ezért az $\alpha_{\sqrt{S}}^*$ különböző kiindulási halmazokhoz ugyanazt az eredményhalmazt is rendelheti. A nem bijektív $\alpha_{\sqrt{S}}$ függvény a kiindulási tényleges állapotokat diszjunkt csoportokra, osztályokra bontja; egy csoportba azok az állapotok tartoznak, amelyekhez a függvény azonos értéket rendel.

▼ **Típusabsztrakció vagy típus, minta vagy séma.** A $\sigma = (\mathcal{T}, S, P)$ és $\sigma_{i \in I} = (\mathcal{T}_i, S_i, P_i)$ rendszerek esetén σ a σ_i rendszerek *típusabsztrakciója* vagy *típusa*, ha létezik minden σ_i rendszerhez olyan $\alpha_{\sqrt{T_i}} : T \rightarrow T_i$ szigorúan monoton növekvő és a tényleges állapotokon értelmezett $\alpha_{\sqrt{S_i}} : \bar{S}_i \rightarrow \bar{S}$ függvény, hogy $\forall i \in I$ és $\forall t \in T$ esetén $P(t) = \alpha_{\sqrt{S_i}}^* (P_i(\alpha_{\sqrt{T_i}}(t)))$, ahol az $\alpha_{\sqrt{S_i}}^* : \bar{S}_i \rightarrow \bar{S}$ az $\alpha_{\sqrt{S_i}}$ kombinációs-függvénye: $\alpha_{\sqrt{S_i}}^* : s \mapsto \bigcup_{\forall x \in s} \{\alpha_{\sqrt{S_i}}(x)\}$. Ha a típusabsztrakciók hasonlítható absztrakciók, akkor σ az egyes rendszerek *mintája*, *kerete* vagy *sémája*.

Példa: Fibonacci-számok.

Az első táblázat egy olyan rendszer első néhány állapotváltását ábrázolja, amely az egymás utáni t időpillanatokban sorban az a adott és e előző Fibonacci-számokat veszi fel értékül. A rendszer komplex állapotát számpárok határozzák meg.

t	a	e
1	1	0
2	1	1
3	2	1
4	3	2

A második táblázat a rendszer egy lehetséges (típus)általánosítását mutatja meg. Az $N \times N$ állapothalmazt az $\{1,2\} \times \{1,2\}$ halmazba képeztük, ahol 1 a páratlan és 2 a páros vagy nulla értéket jelöli. A változás függvényének egy lehetséges megadási módja:

$$s_t = P(t) = \begin{cases} (1, 2) & \text{ha } t \equiv 1 \pmod{3} \\ (1, 1) & \text{ha } t \equiv 2 \pmod{3} \\ (2, 1) & \text{ha } t \equiv 0 \pmod{3} \end{cases}$$

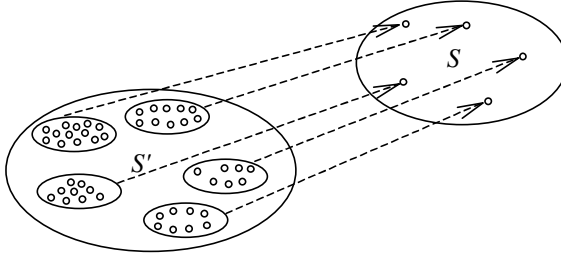
t	a	e
1	1	2
2	1	1
3	2	1
4	1	2
5	1	1
6	2	1

§ Az ekvivalencia a típusabsztrakció speciális esete. Ekkor az $\alpha_{\sqrt{S_i}}$ és $\alpha_{\sqrt{T_i}}$ függvények bijektívek.

§ A típusabsztrakció az absztrakció speciális esete. A típusabsztrakció az alkotó rendszerek szempontjából egy kettős lehetőséget jelent, az *időpontok kiválasztása és átjelölése* mellett az *állapotok átjelölésének és csoportosításának* a lehetőségét. Amennyiben az eredmény nem ekvivalens a kiindulási rendszerekkel, akkor bizonyos $\alpha_{\sqrt{T_i}}$ függvények nem bijektívek, illetve bizonyos $\alpha_{\sqrt{S_i}}$ függvények nem invertálhatók. Ez adott időpontok kiválasztását ($\alpha_{\sqrt{T_i}}(T)$) jelenti, míg állapotok esetén a megfelelő S_i alaphalmazon egy, a megfelelő $\alpha_{\sqrt{S_i}}$ függvény által meghatározott tulajdonságú *osztályozást* indukál. A típusabsztrakció tehát egy időpontban a lehetséges állapotok halmazának csak olyan bővítését engedélyezi, melynél a halmaz minden eleméhez egyértelműen meghatározza, hogy milyen további állapotokat (osztályának további elemeit) kell felvenni a bővebb halmazba. A típusabsztrakció ezért az absztrakció speciális esete.

§ Típusabsztrakció következményhez következményt rendel, azaz ha $\sigma = (\mathcal{T}, S, P)$ a $\sigma' = (\mathcal{T}', S', P')$ rendszer típusabsztrakciója, akkor

$\forall s_1, s_2 \in \bar{S}$ esetén, ha $s_1 \subseteq s_2$, akkor $\alpha^*_{\vee_{S_i}}(s_1) \subseteq \alpha^*_{\vee_{S_i}}(s_2)$, mivel ekkor $s_1 \cup s_+ = s_2$, ami az $\alpha^*_{\vee_{S_i}}$ képzéséből adódóan a következőt eredményezi:
 $\alpha^*_{\vee_{S_i}}(s_2) = \alpha^*_{\vee_{S_i}}(s_1) \cup \alpha^*_{\vee_{S_i}}(s_+)$.



9. ábra. A típusabsztrakció a kiindulási állapotokat osztályokba csoportosítja

Makrorendszerek

A hasonlítható absztrakció és a kompozíció rokon relációk. Mindkettő egy-egy, több rendszerre vonatkozó makrorendszert jelent. A hasonlítható absztrakció általánosított rendszere az elemeit (a többszörös feltételeit) implicit, közvetett módon, közös jellegként foglalja magába. Kompozíció esetén a tartalmazás közvetlen: az összes rész explicit módon, együttesen alkotja az egészt.

A két transzformáció viszonyát az informatika a hasonlítható absztrakció „vagy”, illetve a kompozíció „és” jellegével fogalmazza meg. Ha egy absztrakt rendszer egy adott állapotban van, akkor az azt jelenti, hogy „vagy” az egyik, „vagy” a másik, ... specializált állapotban van. Ezzel szemben egy kompozit komplex állapota esetén az egyik komponens egy adott állapotban van „és” a másik komponens is egy adott állapotban van... és így tovább.

Valójában az „és” jellegű makroállapot a kompozíciónál általánosabb jelentésű és ezt az általánosabb jelentést a többszörös származtatás fogalmában találjuk meg, amelynek a kompozíció csupán speciális esete. Adott rendszerek adott időpontbeli állapotából „és” jelleggel képzett makroállapotnak az egyik „és” a másik állapot által meghatározott feltételeket egyaránt teljesítenie kell.

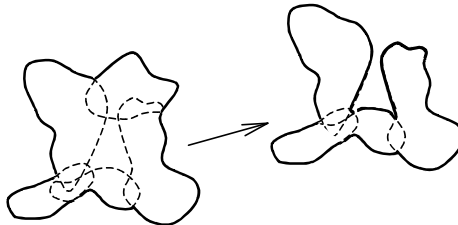
Korlátozás és kiterjesztés

A hasonlítható absztrakciót és a kompozíciót egy fontos tulajdonság kapcsolja össze. Ha egy kompozitból „elhagyjuk” az egyik komponenst, azaz minden időpontban tetszőleges állapotú komponensre cseréljük, akkor egy általánosított rendszert, a kiindulási rendszer következményét, azaz hasonlítható absztrakcióját kapjuk eredményül. Ekkor, a közismert szóhasználatból élve elvonatkoztattunk a kiemelt résztől. Az általánosításnak ezt a formáját *egyszerű absztrakciónak* nevezzük, az ezzel ellentétes átalakítást pedig *egyszerű pontosításnak*.

Egy pontosítás során az eredeti rendszer egy-egy állapotához több lehetséges alállapotot rendelünk. Egyszerű pontosítás esetén az állapotok bővítése például úgy történhet, hogy az absztrakt állapotot egy elempár első elemeként tekintve a második elembe felsoroljuk az aktuális állapotban belüli alváltozatot. Ez az elv valójában a programozásban, illetve tágabb értelemben az információk megadásánál használt mindennapos technika formalizálása, amikor egy általános megfogalmazást a paraméterekkel konkretizálunk.

✓ **Résztől elvonatkoztatás, kivonás vagy egyszerű absztrakció.** Egy σ_C kompozit σ_a és σ_b része esetén a σ rendszer a σ_a és σ_b *különbsége*, vagy σ_a σ_b -től történő *elvonatkoztatása*, ha azonosnak tekinthető a σ_a pontosan azon részeivel, melyek nem σ_b részei. A résztől történő elvonatkoztatás a rendszer következménye, melyet *egyszerű absztrakciónak* nevezzük.

✓ **Rendszer leszűkítése alrendszerére.** A $\sigma = \sigma_1 \otimes \sigma_2$ kompozíció σ_2 -től történő elvonatkoztatását a σ rendszer σ_1 -re történő *leszűkítésének* nevezzük.

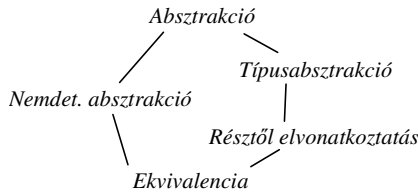


10. ábra. A résztől való elvonatkoztatás egy (egyszerű) absztrakció.

✓ **Kiterjesztés, hozzáadás vagy egyszerű pontosítás.** A $\sigma_1 \otimes \sigma_2$ kompozícióval azonosnak tekintett σ rendszert σ_1 *kiterjesztésének* vagy σ_1 és σ_2 *összegének* nevezzük. A kiterjesztés az eredeti rendszer feltétele.

✓ **Absztrakt rész vagy absztraktum.** Rendszer egy komponensének a következménye a kiindulási rendszer következménye ugyanúgy, mint egy rendszer következményéből egy rész kiválasztása. Tetszőleges következményt ezért *absztrakt résznek*, más néven *absztraktumnak* nevezzük.

☞ Az egyszerű absztrakció típusabsztrakció, a kiterjesztés típuspontosítás.



11. ábra. Az absztrakció típusai.

§ **Pontosítás megadható egyszerű pontosításként.** A $\sigma = (T, S, P)$ kiindulási és a pontosított $\sigma' = (T', S', P')$ rendszer megfelelő átjelölésének kompozíciója ekvivalens lesz a pontosítás eredményével (az állapothalmaz $S'' = S \otimes S'$). Az állítás azt jelenti, hogy *bármely pontosítás eredményével ekvivalens rendszer megadható paraméter-jellegű kiegészítéssel, kiterjesztéssel.*

II

Információ és ismeretszerzés

Rendszer korlátozója a tetszőlegességből kiemeli a rendszert, így alkalmas a rendszerre vonatkozó információ definíciójára. Hasonlítható rendszerek két halmaza alapján definiálhatók az ábrázolással és az ábrázolási eljárással kapcsolatos fogalmak, többek között a „jel” és a „szemantika” fogalmai. Adott rendszerre vonatkozó ismeretek egy-egy „szempont” alapján összehasonlíthatók, mely eltérés mértékként használható. Az ismeretszerzést a vizsgált rendszerre vonatkozó egyre pontosabb ismeretek előállításaként definiáljuk, mely maga is folyamat, azaz ugyancsak rendszerként vizsgálható. A példákon keresztül történő ismeretszerzési folyamat, illetve a kerülőutak egyszerűsítése a rendszert ún. makrorendzerként, egyszerűbb kiindulási rendszerek építőelemeiből képzett együttes ábrázolásként is megadja, amely az ismeretszerzési folyamat menetét követve „megismertet” a rendszerrel.

Információ

Szemléletünkben a rendszer egyetlen megfogható és leírható jellegzetességének a változását tekintjük. A változás függvénye a rendszer minden jellemzőjét magában foglalja, a rendszernek nincsenek általunk vizsgált, ezen kívül eső aspektusai.

A rendszer *korlátozójának* neveztük azon hasonlítható absztrakcióit, amelyek nem tetszőleges absztrakt rendszerek. Minden korlátozó a rendszer megközelítése abban az értelemben, hogy egy ilyen absztraktum bizonyos pontossággal meghatározza az egyes időpillanatokban felvehető lehetséges állapotokat. A korlátozó értelmezett időpontjaiban a bizonytalanságot jelentő teljes állapothalmaztól eltérően szűkíteni, *korlátozni* tudjuk a rendszer által felvehető állapotokat.

A rendszer korlátozója így alkalmas a rendszerre vonatkozó információ definíciójára, mivel a rendszert kiemeli a tetszőlegességből és azt közelítően, vagy éppen egzaktul meghatározza. Hipotézisünk szerint a rendszereken alapuló szemlélet teljes, azaz a világnak nincs a szemléleten kívül eső aspektusa, ezért a korlátozók segítségével a valóságra vonatkozó információk is definiálhatók.

▼ **Információ.** A σ rendszer korlátozója a σ rendszerre vonatkozó *információ* (ekvivalens definíció). A σ rendszerre vonatkozó információk (halmazának) jelölése: $\mathcal{I}_\sigma$.

Megfogalmazásunk nem kerül ellentétbe az információ hagyományos megközelítéseivel sem. Az információelméletben az információ az újdonsággal, a váratlansággal kapcsolatos, így nagyobb információtartalmat tulajdonít egy kisebb valószínűségű esemény bekövetkezésének. Az esemény bekövetkezése információ, mivel a lehetséges változatokból konkretizálva kiválasztja a ténylegesen bekövetkezett ágot.

Az újdonság megfogalmazásához egy kiegészítő definícióra lesz szükségünk. Az újdonság már bizonyos információ birtokában szűkíti tovább a lehetőségeket, így két információ viszonyával kapcsolatos. Egy adott információ szubjektív szempontjából tekintett, a rendszerre vonatkozó információ csak akkor jelent újdonságot, ha annak a már ismert információ nem a példája (nem a hasonlítható pontosítása).

▼ **Szubjektív információ vagy újdonság.** A σ rendszerre vonatkozó σ_I információ a σ rendszerre vonatkozó, annak σ_S hasonlítható absztrakciójának szempontjából vett (*szubjektív*) információ, ha az nem σ_S hasonlítható absztrakciója. A σ -ra vonatkozó, a σ_S szempontjából vett szubjektív információk jelölése: $\mathcal{I}_{\sigma_S}\sigma$.

Példa: Információ és újdonság.

Ha egy személyről a jellemzések megtudjuk, hogy átlagos magasságú, akkor az – a megfelelő környezet „átlagos magasságának” (pl.: 170cm és 190cm közötti magasság) ismeretében – információt jelent a számunkra. Az ezt követő „nem kis növésű” jellemzés már nem számít újdonságnak, ugyanakkor abszolút-értelemben információ, mivel adott

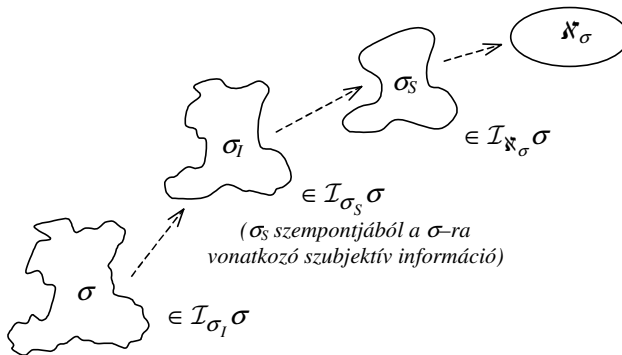
pontosságig ugyancsak meghatározza, a bizonytalanságból kiemeli az illető magasságát. A „nem kis növésű” „absztrakt „rendszer” az „átlagos magasságú” hasonlítható absztrakciója, mivel a „magas növésű” „állapotot” is lehetővé teszi.

A magasság mért értéke az „átlagos magasság” tartományán belüli résztartományt állapít meg, tovább pontosítja a rendszert, ezért újdonságnak számít. Ugyancsak újdonság egy olyan tartomány megadása, amely nem a már ismert tartomány része (pl. 175cm-nél alacsonyabb), de újdonságként jelenik meg egy eddig még ismeretlen jellemző értéke is (pl.: „zömök testalkatú” vagy „barna hajú”).

Az információ és a szubjektív információ maga is rendszer, ezért vizsgálatokor felhasználhatók szemléletünk fogalmai és eredményei.

☞ *A rendszerre vonatkozó információ a rendszer következménye, azaz az információ feltétele a kiindulási rendszer.*

§ Információ és szubjektív információ kapcsolata. A σ' a σ rendszerre vonatkozó információ, ha az a σ hasonlítható tetszőleges absztrakt rendszere ($\aleph_\sigma$) szempontjából vett, a σ rendszerre vonatkozó szubjektív információ.

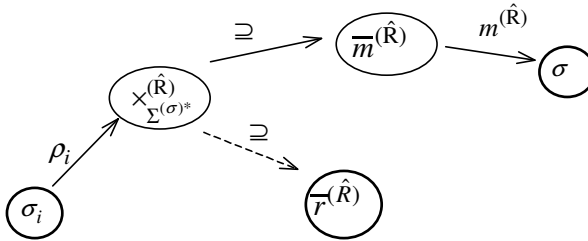


12. ábra: Információ és szubjektív információ.

Ábrázolás

▼ **Hasonlítható rendszerek szorzata, szerep.** A hasonlíthatóvá átjelelt $\sigma_{i \in I}$ rendszerekhez egy $\hat{R}$ indexhalmaz elemeit rendelve a *hasonlítható rendszerek egy szorzatát* kapjuk. Egy σ rendszerrel hasonlítható rendszerek $\Sigma^{(\sigma)*}$ halmazának elemeiből $\hat{R}$ indexhalmazzal képzett szorzatok halmazát $\times_{\Sigma^{(\sigma)*}}^{(\hat{R})}$ -vel jelöljük. Az $\hat{R}$ indexhalmaz ρ elemeit *szerepeknek* nevezzük, mivel meghatározzák az egyes rendszerek szorzaton belül elfoglalt helyeit.

▼ **Módszer, hasonlítható rendszerek közötti reláció.** A $\Sigma^{(\sigma)*}$ hasonlítható rendszerek $\hat{R}$ indexhalmazú szorzatának egy $\bar{m}^{(\hat{R})}$ részhalmazához (azaz $\bar{m}^{(\hat{R})} \subseteq \times_{\Sigma^{(\sigma)*}}^{(\hat{R})}$) hasonlítható rendszert rendelő $m^{(\hat{R})} : \bar{m}^{(\hat{R})} \rightarrow \times_{\Sigma^{(\sigma)*}}^{(\hat{R})}$ függvény a $\Sigma^{(\sigma)*}$ *módszere*. Az $\bar{r}^{(\hat{R})} \subseteq \times_{\Sigma^{(\sigma)*}}^{(\hat{R})}$ halmaz a $\Sigma^{(\sigma)*}$ *hasonlítható rendszerek közötti reláció*.



13. ábra. Módszer és reláció

§ **Módszer és kompozíció.** Kompozit rendszer és a részei hasonlíthatók, ezért a kompozíció módszerként is megadható. A $\sigma = \bigotimes_{i \in I}^{(\pi: I \rightarrow L)} \sigma_i$ kompozíció módszerként történő megadásakor a σ_i rész szerepének választható a $\pi(i) \in L$ helye, azaz $\hat{R} = \pi(I) = L$ (mivel π szürjektív).

§ **Módszer a hasonlítható rendszerek közötti relációvá alakítható.** Egy módszer a hasonlítható rendszerek lehetséges $\hat{R}$ szorzataiból *kiválasztja* az értelmezett szorzatokat, amelyeket egy-egy eredményrendszerré *transz-*

formál. A kiindulási rendszerekkel az eredmény is hasonlítható, ezért minden *módszer* (a kiindulási és az eredmény-) rendszerek közötti relációvá, azaz kiválasztássá alakítható.

Példa: Módszer és reláció.

Vegyük különálló rendszerekként (szituációkként) egy adott időpontban a Föld összes városának nevét. A rendszerek hasonlíthatóak a „szöveg” közös állapothalmaz esetén. Két város között egy reláció megadhatja, hogy azok azonos országhoz tartoznak-e (*AzonosOrszágban(Város, Város)*). A szerepek felcserélhetők, illetve a reláció kiterjeszthető kettőnél több elemre is.

Egy módszer megadhatja egy város esetén, hogy az melyik országban helyezkedik el (*Ország(Város)*). A módszert csak azokon a szövegeken értelmezzük, amelyek ténylegesen városok nevei. A módszer hasonlítható rendszerek közötti relációvá alakítható (*Elhelyezkedik(miben: Ország, mi: Város)*).

Definícióink közül a következő oldalakon elsősorban a módszer fogalmát alkalmazzuk, mint egy olyan függvényt, ami egy-vagy több hasonlítható rendszerhez velük ugyancsak hasonlítható rendszert rendel.

▼ **Absztrakciós és pontosító módszer.** Rendszerhez a hasonlítható absztrakcióját, ill. hasonlítható pontosítását rendelő függvény módszer, amelyet *absztrakciós módszernek* (α), hasonlítható pontosítás esetén *pontosító módszernek* ($\bar{\alpha}$) nevezünk.

Korábbi definíciónk alapján *közvetlen átjelölésnek* nevezzük azt a közvetlenül hasonlítható rendszerek két halmaza közötti $\alpha_{\equiv} : \Sigma^{(\sigma)} \rightarrow \Sigma^{(\sigma')}$ függvényt, amely a kiindulási rendszerhez a rögzített $\alpha_{\equiv_T}$ és $\alpha_{\equiv_S}$ transzformációkkal kapott ekvivalens-rendszert rendeli. Ahogy a rendszerek közvetlen hasonlíthatósága kiterjeszthető hasonlíthatósággá, a közvetlen átjelölésnek is megadható egy általánosabb változata. A (közvetett) átjelölésnél megengedünk egy-egy hasonlíthatóvá történő átjelölést.

▼ **Átjelölés.** Egy $\alpha_{\equiv} : \Sigma^{(\sigma)} \rightarrow \Sigma^{(\sigma')}$ közvetlen átjelölést a $\sigma_{i \in I} \in \Sigma^{(\sigma_i)*}$ hasonlítható rendszerek és a $\sigma'_{j \in J} \in \Sigma^{(\sigma'_j)*}$ hasonlítható rendszerek közötti

(közvetett) átjelölésnek nevezzük (jelölés: $A : \Sigma^{(\sigma)^*} \rightarrow \Sigma^{(\sigma')^*}$), ha a σ_i rendszerek a σ -val, σ'_j rendszerek a σ' -vel hasonlíthatóvá átjelölhetők.

Átjelölés (közvetett átjelölés) segítségével a σ_i -vel hasonlítható rendszerek egy rögzített $\alpha_{\equiv_T}$ és $\alpha_{\equiv_S}$ függvényű ekvivalens transzformációval átjelölhetők úgy, hogy azok a σ'_j rendszerekkel hasonlíthatók. Azonban lehetséges, hogy az átjelölésnek a σ_i vagy σ'_j rendszerekben nem (csak a σ , illetve σ' rendszerekben) értelmezett időpontokra vagy állapotkombinációkra is meg kell adni a megfeleltetést. Tehát a közvetett átjelölés nem feltétlenül definiálható csak a σ_i és σ'_j rendszerek értelmezett időpontjaival és állapotkombinációival.

☞ *Hasonlítható rendszerek értelmezett idő- és értelmezett állapotkombináció-halmazai.* Hasonlítható rendszerek esetén azok értelmezett időpontjainak és értelmezett állapotkombinációinak halmazai egy-egy közös időpont-, illetve állapotkombináció-halmazzá bővíthetők, melyeknek az eredeti megfelelő halmazok mind a részhalmazai. Jelölésekkel: a $\Sigma^{(\sigma_{i \in I})^*}$ hasonlítható rendszerek esetén minden rendszer értelmezett időpontjainak T_i és értelmezett állapotkombinációinak $\bar{S}_i$ halmazai egy közös $T_\Sigma = \bigcup_{i \in I} T_i$ és $\bar{S}_\Sigma = \bigcup_{i \in I} \bar{S}_i$ halmaz részhalmazai, illetve értelmezett állapotok (nem állapotkombinációk) esetén: $\bar{\bar{S}}_\Sigma = \bigcup_{i \in I} \bar{\bar{S}}_i = \bigcup_{x \in \bar{S}_\Sigma} x$. A σ_i rendszerek $\Sigma^{(\sigma_{i \in I})^*}$ hasonlíthatóvá átjelölése úgy történik, hogy rögzítünk az értelmezett időpontok uniójánál bővebb $\mathcal{T}_\Sigma \supseteq T_\Sigma = \bigcup_{i \in I} T_i$ időpont-halmazt, valamint az értelmezett állapotoknál bővebb $\mathcal{S}_\Sigma \supseteq \bar{\bar{S}}_\Sigma$ állapothalmazt, amelyek a hasonlíthatóvá átjelölt rendszerek esetén minden rendszer számára közösek.

§ Absztrakciós módszer alkalmazása bővíti a hasonlíthatóság körét, azaz $\Sigma^{(\sigma_i)^*} \subseteq \Sigma^{(\alpha(\sigma_i))^*}$. A $\Sigma^{(\sigma_{i \in I})^*}$ hasonlítható rendszereken végrehajtott α absztrakciós módszer $\alpha(\sigma_i)$ eredményei egymással és a kiindulási rendszerekkel is hasonlíthatók. Az absztrakciós módszer eredményeként a rendszer értelmezett időpontjainak és értelmezett állapotkombinációinak hal-

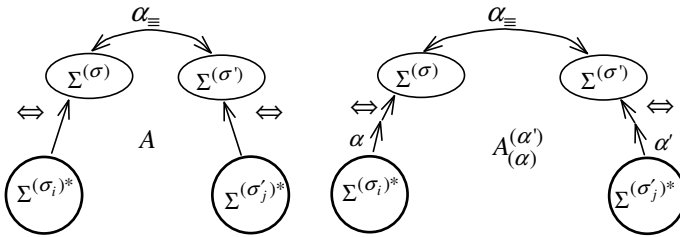
mazai legfeljebb szűkülhetnek, de sohasem bővíülhetnek, azaz $\sigma = (\mathcal{T}, S, P)$ és $\alpha(\sigma) = \sigma' = (\mathcal{T}', S', P')$ rendszerek esetén $\mathcal{T} \supseteq \mathcal{T}'$ és $\bar{S} \supseteq \bar{S}'$. Az értelmezett időpontok és állapotkombinációk szűkebb halmaza a hasonlítható rendszerek nagyobb körét fogja meghatározni, mivel az eredeti halmaz rendszerinél a kevesebb elemet tartalmazó értelmezett időpont és állapotkombináció-halmazokat további rendszerek is tartalmazhatják. Például a tetszőleges absztrakt rendszer bármelyik rendszerrel hasonlítható. Az absztrakciós módszer eredményei tehát hasonlíthatók a kiindulási rendszerekkel, valamint lehetséges, hogy olyan további rendszerekkel is, amelyek a kiindulási rendszerekkel nem hasonlíthatók ($\Sigma^{(\sigma_i)^*} \subseteq \Sigma^{(\alpha(\sigma_i))^*}$). Az eredmények közös idő- és állapotkombináció-halmaza a kiindulási rendszerek közös idő- és állapotkombináció-halmazának a részhalmaza lesz.

Korábban definiált (közvetett) átjelölés fogalmunk nagyon erős kapcsolatot jelent a hasonlítható rendszerek két halmaza között. Egy $A: \Sigma^{(\sigma)^*} \rightarrow \Sigma^{(\sigma')^*}$ átjelölés esetén a transzformációt meg kell adnunk a $\Sigma^{(\sigma)^*}$ rendszerek összes közös T_Σ és $\bar{S}_\Sigma$ elemeire, holott lehetséges, hogy csak bizonyos $\sigma_i \in \Sigma^{(\sigma_i)^*} \supseteq \Sigma^{(\sigma)}$ és hasonlóan képzett $\sigma'_j \in \Sigma^{(\sigma'_j)^*} \supseteq \Sigma^{(\sigma')}$ rendszerek között szeretnénk megteremteni a kapcsolatot. Az átjelölésnek minden (!) $\sigma_x \in \Sigma^{(\sigma)^*}$ rendszernek elő kell állítani annak ekvivalens képét, illetve minden $\sigma_y \in \Sigma^{(\sigma')^*}$ képhez is egy vele ekvivalens kiindulási rendszernek kell létezni.

Általában nincs szükségünk ilyen erős kapcsolatra. Gyakran elegendő csak a kiindulási rendszerek absztrakt képét ábrázolni, illetve általában megengedjük, hogy a kisebb mértékben eltérő ábrázolások, azaz bizonyos ábrázolások absztrakciói ugyanarra a dologra vonatkozzanak. Például a különböző formában kézzel írt vagy éppen nyomtatott Σ jel ugyanazt a dolgot is ábrázolhatja, vagy egy osztály-diagramon más helyen, méretben vagy megjelenési formában ábrázolt osztály ugyanarra a fogalmi vagy akár programozási szinten megjelenő koncepcióra is utalhat.

Az átjelölés, számunkra esetenként szükségtelenül erős kapcsolatát a kiindulási és az eredményrendszereken történő absztrakciós módszer alkalmazásával „absztrakt átjelöléssé” gyengíthetjük.

▮ **Absztrakt átjelölés.** A $\sigma_{i \in I} \in \Sigma^{(\sigma_i)^*}$ hasonlítható rendszerek és azok α absztrakciós módszere, illetve $\sigma'_{j \in J} \in \Sigma^{(\sigma'_j)^*}$ hasonlítható rendszerek és azok α' absztrakciós módszere, valamint egy $\alpha_{\equiv} : \Sigma^{(\sigma)} \rightarrow \Sigma^{(\sigma')}$ átjelölés esetén, amennyiben $\alpha(\sigma_i) \Leftrightarrow \in \Sigma^{(\sigma)}$ és $\alpha'(\sigma'_j) \Leftrightarrow \in \Sigma^{(\sigma')}$ (azaz az $\alpha(\sigma_i)$ rendszerek a σ -val, az $\alpha'(\sigma'_j)$ rendszerek pedig σ' -vel hasonlíthatók), akkor azt mondjuk, hogy a $\Sigma^{(\sigma_i)^*}$ és $\Sigma^{(\sigma'_j)^*}$ rendszerek között megadtunk egy α és α' absztrakciós módszerekkel történő, $\alpha_{\equiv}$ transzformációjú $A_{(\alpha)}^{(\alpha')} : \Sigma^{(\sigma_i)^*} \rightarrow \Sigma^{(\sigma'_j)^*}$ absztrakt átjelölést.



14. ábra: Átjelölés és absztrakt átjelölés.

Az absztrakt átjelölés csak a kiindulási és az eredményrendszerek hasonlítható absztrakciói között teremti meg a kapcsolatot. Az átjelölés megadásakor lehetséges, hogy az kizárólag a rendszerek értelmezett időpontjaira, illetve állapotkombinációira hivatkozik.

Példa: Osztálydiagram és program.

Egy osztálydiagram lehet egy bizonyos, például „Java” programozási nyelven megírt osztályok *absztrakt átjelölése*. A program szövegét több, különböző módon is megírhatjuk, az különböző formátum-konvenciókat (pl. szóközök elhelyezése) és megjegyzéseket használhat, illetve különbözhet a műveleteknél megadott utasítássorozat is. A program szövegéből egy egyszerű absztrakciós módszer le tudja választani a diagram szempontjából lényeges elemeket. Ugyanakkor a diagram is több különböző megjelenési formájú lehet, különböző módokon helyezhetjük el az osztályokat, más betűtípust alkalmazunk, stb., de ezek az eltérések mind egy „absztrakt” diagramra utalnak, így itt is meghatározhatunk egy megfelelő absztrakciós módszert. Mindkét hasonlítható absztrakció típus-absztrakció, mivel az állapotok bizonyos jellemzőitől egyszerűen eltekintünk (elválasztó szóközök, megjegyzések, utasítások, illetve a diag-

ramon az osztályok elhelyezkedése, megjelenési formája, stb.) A két absztrakciós módszer alkalmazásával kapott „lényegi” elemeket vizsgálva ellenőrizhetjük, hogy azok egymás átjelölései-e, azaz a diagram valóban a programra vonatkozik-e és fordítva.

Egy $A_{(\alpha')}^{(\alpha')}$ átjelölés esetén, ha α' a kiindulási rendszerek triviális hasonlítható absztrakciója, azaz minden σ'_j rendszerhez azzal azonosnak tekintett rendszert rendel, az α azonban nem triviális absztrakció, akkor az $A_{(\alpha)}$ absztrakt átjelölés bizonyos σ_i rendszereket csak egy adott absztrakciós szintig képes megközelíteni. Például egy osztálydiagram „elvi” tartalma a programot csak egy adott absztrakciós szintig képes megközelíteni.

Egy $A^{(\alpha')}$ (például $A_{(\alpha)}$ inverze), α' nem triviális és α triviális hasonlítható absztrakciójú absztrakt átjelölés esetén a σ'_j -k a σ_i rendszerek absztrakciós szintjénél pontosabb rendszereket is meg tudnának közelíteni, de ekkor a σ_i rendszerek által már nem értelmezett időpontokra vagy állapotkombinációkra kellene hivatkozniuk.

Ha az α és α' is triviális hasonlítható absztrakció, akkor A egy (közvetett) átjelölés.

Példa: Absztrakt átjelölés.

Egy autó digitális sebességmérője a sebesség tartományaihoz rendel egy-egy számértéket. Ez a sebességmérő a sebességet csak a megfelelő absztrakciós szintig (tartományig) képes megközelíteni ($A_{(\alpha)}$). Egy digitális hangerőszabályzó esetén az egyébként analóg módon változtatható erősítést csak a szabályzó által megadott lépésenként tudjuk módosítani. A szabályzó állapotát a hangerő mértéke „ábrázolja”. A hangerő a szabályzónál finomabb változtatást is képes lenne ábrázolni, de ehhez a szabályzó lehetséges értékén kívüli (köztes) értékeket kellene érzékelnie ($A^{(\alpha')}$).

§ Átjelölés és absztrakt átjelölés. Hasonlítható rendszerek két halmaza közötti *absztrakt átjelölés* (az $\alpha_{\equiv}$ kiterjesztésével) *kiterjeszthető közvetett átjelöléssé*. Ekkor az $\alpha_{\equiv}$ átjelöléshez további időpont- és állapotkombináció-megfeleltetéseket kell definiálni. Az *átjelölés* pedig *absztrakt átjelöléssé szűkíthető*. Ebben az esetben az átjelölés bijekcióit legfeljebb a kiindulási és az eredményrendszerek által értelmezett időpontokra és állapotkombinációk-

ra adjuk meg, illetve megfelelő absztrakciós módszereket választunk. Tehát egy absztrakt átjelölés „háttérben” közvetett átjelölések állnak, egy közvetett átjelölés pedig absztrakt átjelöléseket „tartalmaz”.

Az $A_{(\alpha)}$ absztrakt átjelölés a kiindulási σ_i rendszereket csak egy adott absztrakciós szintig képes megközelíteni. Az absztrakt átjelölésnek ez a fajtája a kiindulási rendszerekhez egy-egy absztrakt eredményt rendel, abban az értelemben, hogy több kiindulási rendszer esetén ugyanazt az eredményt is kaphatjuk, amennyiben a kiindulási rendszerekhez az α is ugyanazt a hasonlítható absztrakciót határozza meg. Az $A_{(\alpha)}$ absztrakt átjelölés így a kiindulási rendszer vázlatos (áttekintő) „átjelölését” állítja elő.

▽ Absztrakt ábrázolási eljárás és módszer. Egy $A_{(\alpha)} : \Sigma^{(\sigma_{i \in I})^*} \rightarrow \Sigma^{(\sigma_{j \in J})}$ absztrakt átjelölést *absztrakt ábrázolási eljárásnak*, $\Sigma^{(\sigma_i, \sigma_j)^*}$ esetén (azaz hasonlítható kiindulási és eredményrendszerek esetén) *absztrakt ábrázolási módszernek* nevezzük.

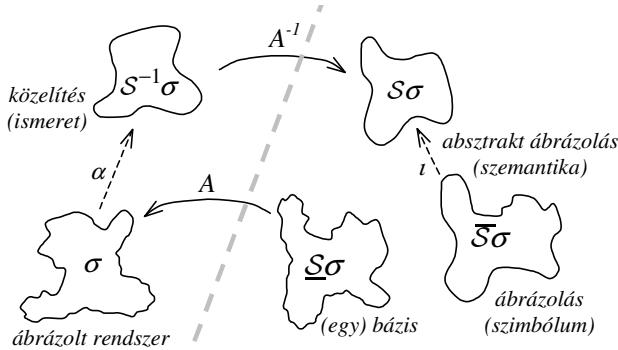
Az „áttekintő”, „absztrakt” képet előállító transzformációnál meg kell adnunk az eredményrendszerek pontos idő és állapothalmazát, azaz rögzítjük az „absztrakt képek” változásának keretét. Ezzel az eredményrendszerek nem csak hasonlíthatók (azaz egy azonosnak tekintett rendszerré közvetlenül hasonlíthatóvá átjelölhetők), hanem „közvetlenül hasonlíthatók”.

▽ Absztrakt ábrázolás. Rögzített $A_{(\alpha)} : \Sigma^{(\sigma_{i \in I})^*} \rightarrow \Sigma^{(\sigma_{j \in J})}$ absztrakt ábrázolási eljárás esetén egy kiindulási $\sigma \in \Sigma^{(\sigma_i)^*}$ rendszernek egy tetszőleges $\sigma' \in \Sigma^{(\sigma_j)}$ rendszer az *absztrakt ábrázolása*.

Mint láthatjuk, egy kiindulási rendszernek bármely, az eredményekkel hasonlítható rendszer lehet az absztrakt ábrázolása, így akár a hasonlítható tetszőleges absztrakt rendszer is. Az absztrakt ábrázolás így nem jelenti automatikusan annak helyes voltát. Azonban absztrakt ábrázolást csak egy rögzített absztrakt ábrázolási eljárás és egy kiindulási rendszer viszonylatában vizsgálhatunk.

▮ **Közelítés vagy ismeret, bázis.** Az $A_{(\alpha)} : \Sigma^{(\sigma_{i \in I})^*} \rightarrow \Sigma^{(\sigma_{j \in J})}$ absztrakt ábrázolási eljárás esetén jelölje $\alpha_{\equiv}$ az $A_{(\alpha)}$ átjelölését, illetve $A : \Sigma^{(\sigma_i)^*} \rightarrow \Sigma^{(\sigma_j)^*}$ ($\Sigma^{(\sigma_j)} \subseteq \Sigma^{(\sigma_j)^*}$) egy, az $A_{(\alpha)}$ kiterjesztéseként kapott tetszőleges (közvetett) átjelölést. A kiindulási $\sigma \in \Sigma^{(\sigma_i)^*}$ rendszer *absztrakt ábrázolásaként* vegyünk egy tetszőleges $\sigma' \in \Sigma^{(\sigma_j)}$ rendszert. Az $\alpha_{\equiv}^{-1}$ átjelölést az adott absztrakt ábrázolásokat *alkalmazó* transzformációnak, annak $\alpha_{\equiv}^{-1}(\sigma')$, azaz $A^{-1}(\sigma')$ eredményét pedig a kiindulási rendszer *közelítésének* vagy *ismeretnek* nevezzük. A kiindulási rendszernek az ábrázolással hasonlítható $A(\sigma) \in \Sigma^{(\sigma_j)^*}$ képe az absztrakt ábrázolás egy *bázisa*.

▮ **Ábrázolás vagy reprezentáció vagy szimbólum, szemantika.** Rögzített $A_{(\alpha)} : \Sigma^{(\sigma_{i \in I})^*} \rightarrow \Sigma^{(\sigma_{j \in J})}$ absztrakt ábrázolási eljárás, valamint egy $(\iota : \Sigma^{(\sigma_j)^*} \rightarrow \Sigma^{(\sigma_j)}, \iota : \sigma'' \mapsto \sigma')$ értelmezésnek nevezett módszer esetén a σ rendszernek σ'' az *ábrázolása*, *reprezentációja*, *jele* vagy *szimbóluma*, ha ahhoz az ι értelmezés $\iota(\sigma'') = \sigma' \in \Sigma^{(\sigma_j)}$ eredményként a kiindulási rendszer absztrakt ábrázolását – a σ'' jel σ' *jelentését* vagy *szemantikáját* – rendeli.



15. ábra: Absztrakt ábrázolás és ábrázolás.

Az ábrázolás definíciója az absztrakt ábrázoláshoz hasonlít, abban az értelemben, hogy ábrázolásnak tekintünk bármely $\sigma' \in \Sigma^{(\sigma_j)^*}$ rendszert,

amelyhez az ι módszer a kiindulási rendszer valamely absztrakt ábrázolást, azaz tetszőleges $\sigma'' \in \Sigma^{(\sigma_j)}$ rendszert rendel.

∇ Absztrakt jel. Vegyünk egy $A_{(\alpha)}$ absztrakt ábrázolási eljárást és egy ι értelmezést. Ha az ι értelmezés invertálható, akkor egy jelhez pontosan egy szemantika, illetve egy szemantikához pontosan egy jel tartozik. Ellenkező esetben minden kiválasztott szemantika ösképe a jelek egy halmaza, azaz a lehetséges jelek (mint rendszerek) absztrakciója. Ezért egy σ' szemantika $\iota^{-1}(\sigma') \subseteq \Sigma^{(\sigma_j)^*}$ ösképét a szemantika *absztrakt jelének* nevezzük.

Az absztrakt jel valójában az azonos értelműnek nevezhető, azaz azonos szemantikájú jelek egy csoportját fogja meghatározni, a tényleges jelek lényegi tartalmát.

Példa: Ábrázolás, absztrakt jel.

Egy számítógép terhelésének százalékos értékét jelenítsük meg (ábrázoljuk) a monitoron. A terhelés változása a számítógép működésének absztrakciója, mivel közös halmazba sorolja azokat az állapotokat, amelyekhez azonos terhelési érték tartozik. Az értéket különböző betűtípussal, mérettel vagy egyéb jellemzőkkel is megjeleníthetjük, melyek (halmaza) egy-egy értékre, az absztrakt jelre utal.

∇ Ábrázolási eljárás. Egy $A_{(\alpha)} : \Sigma^{(\sigma_{i \in I})^*} \rightarrow \Sigma^{(\sigma_{j \in J})}$ absztrakt ábrázolási eljárás és egy $\iota : \Sigma^{(\sigma_j)^*} \rightarrow \Sigma^{(\sigma_j)}$ értelmezési módszer inverzének függvény-kompozíciója minden kiindulási σ rendszerhez annak $\iota^{-1}(A_{(\alpha)}\sigma)$ absztrakt jelét rendel. A $\bar{\beta}(\iota^{-1}(A_{(\alpha)}\sigma))$ összetett függvényt *ábrázolási eljárásnak*, $\Sigma^{(\sigma_i, \sigma_j)^*}$ esetén *ábrázolási módszernek* nevezzük, ahol a $\bar{\beta}$ pontosító módszer minden egyes absztrakt jelhez pontosan egy adott jelet rendel. Az $A_{(\alpha)}$ absztrakt ábrázolási eljárás és egy ι értelmezési módszer több, különböző $\bar{\beta}$ függvényű ábrázolási eljárást ad meg, azaz ábrázolási eljárásokat csak az absztrakt jel erejéig határoz meg. Az $A_{(\alpha)}^{(\iota)} = (A_{(\alpha)}, \iota)$ módon jelölt elemkettes ezért egy $A_{(\alpha)}$ és ι által megadott (de tetszőleges $\bar{\beta}$ pontosító módszerű) ábrázolási eljárást határoz meg.

Az ábrázolási eljárás ezért meghatározza a kiindulási rendszer absztrakt jelét, valamint azok közül kiválaszt egy konkrét jelet, azaz egy tényleges ábrázolást.

§ Az átjelölés egy ábrázolási eljárás. Ekkor az értelmezés módszere a triviális hasonlítható absztrakció (amely a kiindulási rendszerrel azonosnak tekintett rendszert eredményez).

✓ **Egyszerű ábrázolás.** Az $A_{(\alpha)}^{(\alpha')}$ ábrázolást, azaz az absztrakt átjelölést *egyszerű ábrázolásnak* nevezzük. Ekkor az értelmezés egy α' absztrakciós módszer, amely egy ábrázoláshoz szemantikaként annak hasonlítható absztrakcióját rendeli.

Szempont, pontosság és eltérés

Egy kiindulási rendszer ábrázolása mögött szemantikaként egy absztrakt ábrázolás húzódik meg. Absztrakt ábrázolás azonban bármely, az eredmény-rendszerekkel hasonlítható rendszer lehet, így például a tetszőleges absztrakt rendszer is. Az absztrakt ábrázolás és a kiindulási rendszer összevetésére két módunk is van. Egyrészt az ábrázolás mögötti ismeretet összehasonlíthatjuk a kiindulási rendszerrel. Korábbi definíciónk alapján az ismeret az absztrakt ábrázolásra alkalmazott, az absztrakt átjelölés inverzének képe, amely a kiindulási rendszerrel hasonlítható. Másik lehetőségünk, hogy az absztrakt ábrázolást összehasonlítsuk a kiindulási rendszer (bármely) bázisával.

Mivel minden ábrázolási eljárás mögött valójában egy absztrakt ábrázolási eljárás áll, ezért az absztrakt ábrázolás és a kiindulási rendszer összevetése kiterjeszthető általában az ábrázolásokra is.

✓ **Egzakt, közelítő és hibás ábrázolás és ismeret.** Rendszer *egzakt* vagy *pontos ábrázolásának* szemantikája azonosnak tekinthető az ábrázolás (bármely) bázisával (ekkor a bázisok azonosnak tekinthetők). Rendszer *közelítő ábrázolásának* szemantikája bármely bázis nem triviális hasonlítható absztrakciója. A nem egzakt és nem közelítő ábrázolás a *hibás ábrázolás*. Hasonlóan definiálható az *egzakt, közelítő* és *hibás* ismeret (más néven: *közelítés*) is, a kiindulási rendszer és az ismeret összevetésével.

Definíciónk segítségével meghatározhatjuk a kiindulási rendszer és az ismeret vagy az ábrázolás viszonyát. Az ismeretek finomabban minősíthetők, illetve összehasonlíthatók, ha nem csak azok viszonyát, hanem azok eltérését is meg tudjuk határozni. Két, hasonlítható rendszer egymástól való eltérése azonban különböző *szempont* esetén más és más is lehet.

▼ **Szempont vagy nézőpont.** A $\sigma_{j \in J}$ hasonlítható rendszerek $\prec$ (ejtsd: „v”) *szempontja* vagy *nézőpontja* egy i_0 -al jelölt minimumú és i_∞ -el jelölt maximumú rendezett halmazon értelmezett, azokhoz a $\Sigma^{(\sigma_j)^*}$ absztrakciós módszereit rendelő függvény, mely az i_0 értékhez a triviális hasonlítható absztrakciót, i_∞ értékhez pedig egy hasonlítható tetszőleges absztrakt rendszert eredményező absztrakciót rendel, valamint minden $a \geq b$ ($a, b \in D_\prec$) és $\sigma \in \Sigma^{(\sigma_j)^*}$ esetén a $\prec_{(a)}(\sigma)$ a $\prec_{(b)}(\sigma)$ hasonlítható absztrakciója (azaz következménye).

A szempont tehát egy olyan függvény, amely a kiindulási rendszerhez vele hasonlítható, egyre absztraktabb és absztraktabb képet rendel. Szempontot kapunk például akkor, ha először a kiindulási rendszeren, majd a közbülső eredményeken egyre újabb és újabb absztrakciós módszert hajtunk végre, ahogy azt a következő állítás mutatja:

§ **Absztrakciós módszerek sorozata, mint szempont.** A $\sigma_{j \in J}$ hasonlítható rendszereken értelmezett α_i absztrakciós módszerek sorozata, ahol α_0 triviális hasonlítható absztrakció, a $\Sigma^{(\sigma_j)^*}$ egy szempontját határozzák meg, ha a $\prec_{(i)}$ értéke az $\alpha_0 \circ \alpha_1 \circ \dots \circ \alpha_{i-1} \circ \alpha_i$ összetett függvény, amely határértéke egy hasonlítható tetszőleges absztrakt rendszert eredményező absztrakció.

A szempont fogalmának felhasználásával már definiálható a rendszerek *eltérése*, mégpedig úgy, hogy a rendszereken végrehajtva az absztrakciós módszereket, megkeressük azt a legkisebb értéket, amely már azonosnak tekintett rendszereket fog eredményezni. A rendszerek *eltérése* tehát egy index-érték (azaz érték) lesz, amely azonos szempont esetén más eltéréssel is

összehasonlítható. Az eltérésnek megfelelő azonosnak tekintett rendszert a kiindulási rendszerek *pontosságának* nevezzük.

▼ **Rendszerek eltérése, rendszerek pontossága.** A $\sigma_{j \in J}$ hasonlítható rendszerek adott $\prec$ szempont szerinti *eltérése* (jelölés: $\Delta_{j \in J}^{(\prec)} \sigma_j$) az a legkisebb olyan $d \in D_{\prec}$ érték, melyre minden $j \in J$ esetén a $\prec_{(d)}(\sigma_j)$ rendszerek azonosnak tekinthetők. A $\sigma_{j \in J}$ rendszerek $\prec$ szempont szerinti *pontossága* (jelölés: $\lfloor \sigma_{j \in J} \rfloor_{\prec}$) a $\prec_{(d)}(\sigma_j)$ rendszer.

☞ *A pontosság maximális eltérés esetén a tetszőleges absztrakt rendszer.*

☞ *Hasonlítható rendszerek pontossága hasonlítható bármely szempont esetén.*

A pontosság a kiindulási rendszerekkel nem csak hasonlítható, hanem azokkal nagyon szoros kapcsolatban is van:

☞ *A pontosság a kiindulási rendszerek következménye (hasonlítható absztrakciója) bármely szempont esetén.*

☞ *Hasonlítható rendszerek esetén létezik a pontosságuk bármely szempont esetén, mivel az a $\mathfrak{K}_{\sigma}$ hasonlítható tetszőleges absztrakt rendszer példája.*

☞ *Azonosnak tekinthető rendszerek eltérése minimális (azaz $i0$), pontosságuk pedig a rendszerekkel szintén azonosnak tekinthető bármely szempont esetén.*

☞ *Párok távolsága esetén a $\Delta^{(\prec)}(\sigma_a, \sigma_c)$ legfeljebb $\Delta^{(\prec)}(\sigma_a, \sigma_b)$ és $\Delta^{(\prec)}(\sigma_b, \sigma_c)$ maximumértéke. Ha σ_a -val jelöljük a σ_b -től nagyobb vagy egyenlő távolságú rendszert, akkor a $\lfloor \sigma_a, \sigma_b \rfloor_{\prec}$ pontosság (ami $= \lfloor \sigma_a, \sigma_b, \sigma_c \rfloor_{\prec}$) egyaránt a σ_a , σ_b és σ_c hasonlítható absztrakciója, így a $\lfloor \sigma_a, \sigma_c \rfloor_{\prec}$ hasonlítható absztrakciója is.*

A szempont értelmezési tartománya tetszőleges rendezett halmaz lehet. Amennyiben a kiterjesztett nem-negatív valós számok halmazát, vagy annak egy megfelelő részhalmazát választjuk, akkor a hasonlítható rendszerek körében a szempont szerinti eltérés egy távolságot ad meg, így beszélhetünk a

rendszerhez (a szempont szerint) jobban vagy kevésbé hasonló rendszerek-ről!

§ Rögzített kiterjesztett valós szempont szerinti eltérés távolság a hasonlítható rendszerek halmazán az azonosnak tekinthető rendszerek ekvivalencia-osztályain (a szempont értelmezési tartománya a nemnegatív valós számok kiegészítve a ∞ értékkel: $i0 = 0$ és $i\infty = \infty$), mivel

1. $\Delta^{(<)}(\sigma_a, \sigma_b) \geq 0$ és $=0$ pontosan akkor, ha $\sigma_a \Leftrightarrow \sigma_b$,
2. $\Delta^{(<)}(\sigma_a, \sigma_b) = \Delta^{(<)}(\sigma_b, \sigma_a)$, valamint
3. $\Delta^{(<)}(\sigma_a, \sigma_b) + \Delta^{(<)}(\sigma_b, \sigma_c) \geq \Delta^{(<)}(\sigma_a, \sigma_c)$ (mivel $\Delta^{(<)}(\sigma_a, \sigma_c)$ legfeljebb $\Delta^{(<)}(\sigma_a, \sigma_b)$ és $\Delta^{(<)}(\sigma_b, \sigma_c)$ maximuma).

§ Következmények pontossága. A $\sigma_a \Rightarrow \sigma_b \Rightarrow \sigma_c$ rendszerek esetén $|\sigma_a, \sigma_b| \models |\sigma_a, \sigma_c| \Leftrightarrow |\sigma_a, \sigma_b, \sigma_c|$.

A rendszerek eltérésének, illetve pontosságának az eszközeivel már az ismeretek és az ábrázolások pontossága is definiálható.

▽ Ábrázolás és ismeret pontossága. Egy σ rendszer $A_{(\alpha)}^{(t)}$ ábrázolási eljárású σ'' ábrázolásának adott $\prec$ szempont szerinti pontossága az $|A(\sigma), \iota(\sigma'')|_{\prec}$, azaz az ábrázolás (bármely) bázisának és a szemantikának (a megfelelő absztrakt ábrázolásnak) a pontossága. (Az A az $A_{(\alpha)}$ tetszőleges kiterjesztése.) Az ismeret (közelítés) pontossága az ábrázolt rendszer és az ismeret pontossága $(|\sigma, A^{-1}(\iota(\sigma''))|_{\prec})$. Ábrázolás és ismeret pontossága maga is rendszer, így beszélhetünk adott szempont szerinti pontosabb vagy absztraktabb ábrázolásról, illetve közelítésről.

☞ *Ábrázolás szemantikájának következménye az ábrázolás pontossága bármely szempont esetén.*

☞ *Egzakt és közelítő ábrázolás pontossága azonosnak tekinthető az ábrázolás szemantikájával bármely szempont esetén.*

☞ *Hibás ábrázolás pontossága* a szemantika nem triviális hasonlítható absztrakciója.

☞ *Egzakt ábrázolás pontosabb* bármely nem egzakt ábrázolásnál bármely szempont esetén.

☞ *Ismeret pontossága – mint ismeret – egy nem hibás ismeret* bármely szempont esetén.

☞ *Függő és nem hibás (egzakt vagy közelítő) ismeretek esetén, ha az egyik ismeret egy adott szempont szerint a többinél pontosabb, akkor az a többi feltétele (definíció szerint a függő ismeretek feltétel vagy következmény viszonyban vannak egymással).*

§ Egzakt vagy közelítő függő ismeretek pontossága. Függő és nem hibás ismeretek esetén, ha az egyik ismeret a többinél pontosabb egy adott szempont szerint, akkor bármely szempont szerint legalább olyan pontos, mint a többi ismeret.

☞ *Nem hibás ismeretek nem lehetnek ellentmondó rendszerek.*

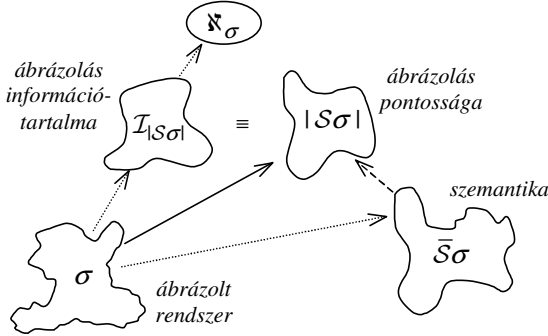
§ Egzakt vagy közelítő és nem függő ismeretek pontossága. Nem hibás és nem függő (tehát független) ismeretek esetén létezik olyan szempont, melyben az egyik, és létezik olyan, amelyben egy másik ismeret a pontosabb. Nem hibás és nem függő ismeretek esetén az ismeretek konjunkciója – mint ismeret – bármely szempont esetén legalább olyan pontos, mint a többi ismeret.

Példa: Közelítő és hibás ábrázolás.

Az autó digitális sebességmérője *közelítően* ábrázolja a sebességet. Ha egy hiba folytán a sebességhez tartozó értéknél eggyel nagyobb értéket jelez ki, akkor az ábrázolás *hibás*. Ekkor azonban még az értékekhez tartozó sebesség-tartományok megfelelő bővítésével, azaz a hibás mérés *pontosságának* meghatározásával a sebességről információt kaphatunk.

Ahhoz, hogy egy rendszer ábrázolásából rendszerre vonatkozó információt nyerjünk, alapvetően három transzformációt kell végrehajtanunk. Első lépésben, az *értelmezéssel* megadjuk a szimbólum szemantikáját, majd egy hasonlítható absztrakcióval meg kell határoznunk az ábrázolás pontosságát. Ezután a pontosság közvetett átjelölésével, *alkalmazásával* a kiindulási rendszerrel hasonlítható rendszert állítunk elő, amely – nem ellentmondó ábrázolás esetén – a rendszerre vonatkozó információ lesz, azaz bizonyos

időpontokban szűkíteni tudjuk a rendszer által felvehető lehetséges állapotok halmazát. A transzformációkkal (két *módszer* és egy *átjelölés*), amelyeket együttesen az *ábrázolás értelmezésének* nevezhetünk, az ábrázolás rendszerre vonatkozó információtartalmát határoztuk meg.



16. ábra: Ábrázolás szemantikája, pontossága és információtartalma.

✓ Ábrázolás és ismeret információtartalma. Rendszer ábrázolásának és egyben az ismeretnek az információtartalma az ábrázoláshoz tartozó ismeret (közelítés) pontossága. Ha a σ kiindulási rendszernek a közelítés nem az ellentéte, valamint a közelítés nem egy tetszőleges absztrakt rendszer, akkor az ábrázolás és az ismeret információtartalma a σ -ra vonatkozó információ. Ellenkező esetben (például ha a σ egy tetszőleges absztrakt rendszer), akkor a pontosság nem lesz a rendszerre vonatkozó információ.

Ha egy függvénnyel egy kiválasztott szempont alapján a rendszerekhez egy rendezett halmaz egy-egy elemét rendeljük, akkor a rendszereken egy rendezést definiálunk. A rendezés megtarthatja a következmény-viszonyokat, azaz rendszer következményéhez a rendszerhez rendelt értéknél nagyobb értéket rendelhet, amelyet *következmény-rendezésnek* nevezünk.

Rögzített szempont esetén egy kiválasztott (kiindulási) rendszerre vonatkozóan a hasonlítható rendszerek (ismeretek) eltérése a rendszerek egy következmény-rendezését határozza meg.

✓ Hasonlítható rendszerek szempont szerinti rendezése. A $\sigma_{j \in J}$ hasonlítható rendszerek $D_{f_{(<)}} \subseteq \Sigma^{(\sigma_{j \in J})^*}$ részhalmazának egy rögzített $\prec$

szempont szerinti rendezésén egy $f_{(\prec)} : D_{f_{(\prec)}} \rightarrow D_{\prec}$ függvényt értünk, amely minden rendszerhez a szempont értelmezési tartományának (az $[i0, i\infty]$ zárt intervallum) egy értékét rendeli.

▽ Hasonlítható rendszerek következmény-rendezése. A $\sigma_{j \in J}$ hasonlítható rendszerek $f_{(\prec)}$ rendezése *következmény-rendezés*, ha bármely $\sigma_a \Leftarrow \sigma_b$ rendszer esetén minden $\prec'$ szempontra $f_{(\prec')}(\sigma_a) \geq f_{(\prec')}(\sigma_b)$.

§ Rögzített rendszertől való eltérés egy következmény-rendezés. Egy rögzített σ rendszer esetén a $\sigma' \in \Sigma^{(\sigma)*}$ rendszerekhez egy kiválasztott szempont alapján a σ rendszertől vett eltérésüket rendelve következmény-rendezést kapunk. Függő rendszerek esetén ezért beszélhetünk egy adott rendszertől jobban, azonos mértékben, illetve kevésbé eltérő rendszerről.

☞ Következmény-rendezés a hasonlítható tetszőleges absztrakt rendszerhez maximális értéket rendel, így a rendezés az adott szempont alapján egyetlen más rendszerhez sem rendelhet nagyobb értéket.

Ismeretszerzés

Egy informatikus mindennapos feladatát, egy program vagy egy információs rendszer szervezésének, tervezésének és programozásának egymásutáni lépéseit egy adott követelményeket teljesítő rendszer ismételt közelítéseként fogalmazhatjuk meg. Ha ez a közelített rendszer a valóság egy részlete, akkor ez a megfogalmazás magát a tanulási folyamatot írja le. Így mondhatjuk azt, hogy egy adott, de pontosan meg nem fogalmazott követelményrendszert kielégítő rendszer fokozatos közelítése, körvonalainak egyre pontosabb megrajzolása valójában a rendszerre vonatkozó *ismeretek szerzése*.

▽ Monoton közelítés. A σ esetén az azzal hasonlítható $\sigma_{i \in I}$ (ahol I egy $i0$ -al jelölt minimumú rendezett indexhalmaz) ismeretek a σ *monoton közelítését* alkotják, ha minden σ_i feltétele σ , valamint $\forall i < j$ ($i, j \in I$) esetén σ_i -nek feltétele a σ_j .

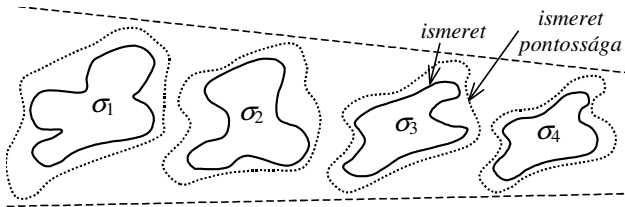
▽ Kontrakció. Adott σ rendszer $\sigma_{i \in I}$ monoton közelítése *kontrakció*, ha létezik olyan $i \in I$, hogy σ_i a σ_{i0} nem triviális következménye.

A monoton közelítés a kiválasztott σ rendszer egyre pontosabb képeit állítja elő vagy esetleg a korábbi képekkel azonosnak tekintett ismereteket. Kontrakció esetén megköveteljük, hogy a kiindulási ismeretnél a monoton közelítés legalább egy pontosabb ismeretet is előállítson.

▽ Ismeretszerzés vagy tanulás. Adott σ rendszer és egy $\prec$ szempont esetén vegyük a (rendszerrel hasonlítható) $I \rightarrow \Sigma^{(\sigma)^*}$ ismereteket, ahol I egy $i0$ -al jelölt minimumú, (legalább kételemű) rendezett indexhalmaz. Az $i \in I$ értékhez tartozó $p_i^{(\prec)}$ jelölje az $i0 \leq j \leq i$ indexekhez rendelt ismeretek $\prec$ szempont szerinti pontosságának konjunkcióját, a *tanulás pontosságait*. Az $I \rightarrow \Sigma^{(\sigma)^*}$ hozzárendelés a $\prec$ szempont *ismeretszerzése*, vagy *tanulása*, ha az $i \mapsto p_i^{(\prec)}$ egy σ -ra vonatkozó kontrakció.

Az ismeretszerzéstől tehát azt várjuk el, hogy közelítéseivel a rendszer pontosabb körvonalait határozza meg.

Az ismeretszerzés is folyamat, így az rendszerként vizsgálható.



17. ábra. A tanulás a rendszer pontosabb körvonalait határozza meg.

Az ismeretek pontosságai kontrakciót alkotnak, de előfordulhat, hogy az ismeretszerzés bizonyos időszakokon (index-tartományokon) belül „leáll”, és csak triviálisan pontosabb, azaz a korábbiakkal azonosnak tekintett pontossággal közelíti meg a rendszert.

A kontrakciótól megköveteljük, hogy minden rendszer, minden „ismeret” a közelítendő rendszer következménye, annak hasonlítható absztrakciója legyen. Ismeretszerzés esetén ennél több engedményt teszünk, mindössze az

ismeretek pontosságainak kell kontrakciót alkotniuk. Itt tehát a folyamatban lehetséges a közelítendő rendszertől eltérő (független vagy akár ellentmondó) ismeret is, mivel mindössze azok pontosságainak viszonyait vizsgáljuk. Például, a rendszerről egy függő ismeret előállítása is lehet tanulás, ha annak pontossága a kiindulási ismerettől kevésbé tér el.

Hőmérsékleti értékek esetén például a kezdeti $[20^{\circ}; 24^{\circ}]$ tartományt követő $[22^{\circ}; 25^{\circ}]$ „ismeret” a tényleges $[22^{\circ}; 23^{\circ}]$ tartományra vonatkozó ismeretszerzés lesz. Ekkor a kezdeti szituáció pontossága $[20^{\circ}; 24^{\circ}]$, a következő mérés pontossága $[22^{\circ}; 25^{\circ}]$, melynek a korábbi ismeret pontosságával vett konjunkciója $[22^{\circ}; 24^{\circ}]$, így a pontosságok ismételt konjunkciói kontrakciót alkotnak.

A kontrakció éppen ezért egy szempontoktól független „ideális ismeretszerzésként” jelenik meg, amely soha nem tér el a kiindulási rendszertől, hanem annak a többé-kevésbé bővebb képét állítja elő.

§ A kontrakció egy ismeretszerzés bármely szempont szerint, amelyben minden előállított ismeret pontossága önmaga, illetve az ismeretszerzés pontossága (a korábbi ismeretek pontosságának konjunkciójaként előálló ismeretek) is a kontrakció ismeretei.

A következőkben elsősorban a lépésenkénti $(\sigma_0, \dots, \sigma_n)$ ismeretszerzésekre helyezjük a hangsúlyt.

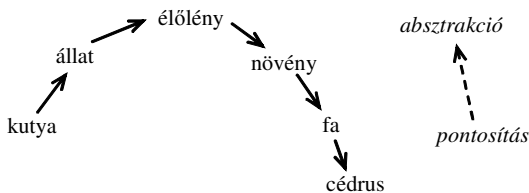
Példákon keresztül történő ismeretszerzés

Az ismeretszerzéshez szükséges az ismeret tesztelése, hogy az mennyiben egyezik meg, illetve mely részleteiben tér el a közelíteni kívánt rendszertől (összehasonlítás), valamint az ismeretnek az eltérés alapján történő módosítása. Általában sajnos egyetlen lépésben nem tudjuk megismerni a vizsgált rendszert. Bonyolult rendszerek közelítését ezért csak egyes eleminek, zártnak, „ismertnek” tekintett részletek összeépítésével, az elemek – részleges tudásunkkal és áttekinthetőségünkkel megfogható – összekapcsolásával tudjuk megadni. Összetett ismeretek ábrázoláshoz így az is szükséges, hogy egyes eleminek tekintett rendszereket, nagyobb, bonyolultabb, összetettebb rendszerré, azaz makrorendszerré kapcsolhassunk össze.

„Ismert” kiindulási rendszerekből adott, „ismert” módon építkezünk, ezért az eredményül kapott rendszer is „ismert” lesz. Így, ha az ismeretünk a közelíteni kívánt rendszerrel azonosnak tekinthető módon változik, akkor a keresett rendszert megismertük.

Egy rendszer (jelöljük σ -val) megismerésének kiindulópontjául ismeretként bármely, azzal hasonlítható rendszert kiválaszthatunk. A hasonlítható tetszőleges absztrakt rendszer ($\aleph_\sigma$) jó kiindulópont, mivel az biztosan a közelítendő rendszer absztrakciója. Az ábrázolás módosításához elegendő egyetlen hasonlítható absztrakció vagy hasonlítható pontosítás a következők alapján.

- Ha az ismeret túlságosan *absztrakt*, azaz a közelítendő rendszer következménye (hasonlítható absztrakciója), akkor hajtsunk végre egy hasonlítható pontosítást, azaz helyettesítsük egy példájával.
- Ha az ismeret túlságosan pontos, így a közelítendő rendszernek mindössze a *példája*, azaz nem tartalmazza az összes lehetséges példát, akkor hajtsunk végre egy hasonlítható absztrakciót, célszerűen olyat, amely a hiányzó példákat is tartalmazza.
- Ha egyik sem a másik következménye, azaz a rendszerek nem függők, akkor hajtsunk végre egy (nem triviális) hasonlítható absztrakciót, célszerűen olyat, amely eredménye az ismeret mellett a közelítendő rendszer hasonlítható absztrakciója is lesz. Létezik ilyen absztrakció, mivel bármely rendszernek a hasonlítható tetszőleges absztrakt rendszer a következménye.
- Ha az ismeret egzakt, azaz a közelítendő rendszerrel azonosnak tekinthető, akkor a rendszert megismertük.



18. ábra. Találgatás: a megfejtés megismerése absztrakcióval és pontosítással.

Az ismeret módosításának egy lépéséhez tehát elegendő egy hasonlítható absztrakció vagy egy hasonlítható pontosítás. Egy transzformációs lépés az absztrakció/pontosítás iránya mellett megadható egy-egy eleminek, „ismertnek” tekintett rendszer kiegészítő felvételével. Az ismeretek ábrázolása során nem célszerű megkülönböztetni az azonosnak tekintett ($\Leftrightarrow$) rendszereket. Ezért a szükséges eszközöket tovább szűkíthetjük egy hasonlíthatóvá történő átjelölésre, valamint a rendszerek uniójára és metszetére.

A közelítés *absztrakciós irányban* történő változtatását az ismerettel azonosnak tekintett rendszer és egy kiegészítő rendszer uniójával adjuk meg. Korábbi állításunk alapján minden hasonlítható absztrakcióhoz létezik ilyen kiegészítő rendszer, például ilyen maga az eredményül kapott rendszer is.

A közelítést az elemi lépésekkel úgy *pontosíthatjuk*, hogy vesszük az ismerettel azonosnak tekintett rendszer és egy kiegészítő rendszer metszetét. A művelet a megfelelő kiegészítő rendszerrel az ismeret tetszőleges hasonlítható pontosítását előállítja. Ilyen kiegészítő rendszer például maga az eredményül kapni kívánt ismeret is.

Tehát az ismeret módosítható egy „ismertnek” feltételezett *rendszerrel történő kiegészítéssel*, amely módosítást megadhatjuk egy esetleges hasonlítható átjelölés mellett az unió és a metszet elemi műveleteivel. Vizsgáljuk meg, hogy az unió vagy metszet művelet során hogyan változik a közelítés!

Absztrakciós irány esetén ($\sigma_{i+1} = \sigma_i^{\Leftrightarrow} \cup \sigma_{+i}$), ha az ismeret a kiegészítő rendszer hasonlítható absztrakciója ($\sigma_i \Rightarrow \sigma_{+i}$), akkor nem történik változás, az ismeret „elnyeli” a kiegészítő rendszert ($\sigma_{i+1} = \sigma_i^{\Leftrightarrow}$). Ha a kiegészítés az ismeret hasonlítható absztrakciója ($\sigma_i \Leftarrow \sigma_{+i}$), akkor a kiegészítés „nyeli el” az ismeretet, tehát ezután maga a kiegészítés lesz az új ismeret ($\sigma_{i+1} = \sigma_{+i}$). Ha a közelítés és a kiegészítés nem függő rendszerek, akkor az új ismeret az unió lesz, amely az eredeti közelítés nem triviális hasonlítható absztrakciója. Az absztrakciós irányban történő kiegészítés eredménye tehát az ismeret hasonlítható absztrakciója, azaz következménye lesz.

Ha az absztrakciós irányban történő kiegészítést az ismeret (azzal közvetlenül hasonlítható) ellentétével hajtjuk végre, akkor azzal „töröljük” az eddigi ismeretet és eredményül a hasonlítható tetszőleges absztrakt rendszert kapjuk. A *rendszer ellentétének* képzését mindössze jelölésként, rövidítésként használjuk fel, mivel egy ismert rendszer ellentéte is ismert.

Pontosítás iránya esetén ($\sigma_{i+1} = \sigma_i^{\Leftrightarrow} \cap \sigma_{+i}$), ha a kiegészítés az ismeret hasonlítható absztrakciója ($\sigma_{+i} \Leftarrow \sigma_i$), akkor nem történik változás, az ismeret „elnyeli” a kiegészítést ($\sigma_{i+1} = \sigma_i^{\Leftrightarrow}$). Ha az ismeret a kiegészítő rendszer hasonlítható absztrakciója ($\sigma_i \Rightarrow \sigma_{+i}$), akkor a kiegészítés „nyeli el” az ismeretet, tehát ezután maga a kiegészítés lesz az ábrázolás ($\sigma_{i+1} = \sigma_{+i}$). Ha a közelítés és a kiegészítés függetlenek, akkor az új ismeret a két rendszer magrendszere, amely eredménye az eredeti közelítés egy nem triviális hasonlítható pontosítása lesz. A közelítés ellentmondó rendszerrel történő kiegészítő pontosítását nem értelmezzük, ez ugyanis egy önmagában ellentmondásos, paradox rendszert eredményezne. A pontosítás irányában történő (nem ellentmondó, értelmezett) kiegészítés eredménye tehát az ismeret hasonlítható pontosítása, azaz feltétele lesz.

A σ_i közelítésnek egy σ_{+i} kiegészítő rendszerrel történő absztrakció ($\cup$) vagy pontosító irányban ($\cap$) történő módosítása esetén a következő speciális esetek fordulhatnak elő:

ismeret törlése	$\sigma_i^{\Leftrightarrow} \cup \sigma_{+i}$	ha $\sigma_i^{\Leftrightarrow}$ ellentéte σ_{+i}
ismeret cseréje absztraktabbra	$\sigma_i^{\Leftrightarrow} \cup \sigma_{+i}$	ha σ_i következménye σ_{+i}
ismeret cseréje pontosabbra	$\sigma_i^{\Leftrightarrow} \cap \sigma_{+i}$	ha σ_i feltétele σ_{+i}

Mindhárom esetben az eddigi közelítés „elveszik”, azt egy új rendszerrel helyettesítjük, amely vagy (törlés esetén) a hasonlítható tetszőleges absztrakt rendszer, vagy az absztraktabb vagy pontosabb kiegészítő rendszer.

Az ismeret törlése, absztraktabbra, pontosabbra, illetve – egy törlést követő pontosítással – tetszőleges rendszerre történő cseréje mind kifejezhető az elemi műveletekkel. A műveletek azonban az ismeret finomabb módosítását is lehetővé teszik. Ha a megismerendő rendszerben találunk egy olyan példát, amelynek nincs megfelelője a közelítésben, akkor az ismeretünk nem eléggé absztrakt. A példával – mint kiegészítő rendszerrel – a közelítés az absztrakciós irányban úgy módosítható, hogy az már tartalmazza a lehetséges példát. Ha az ismeretben találunk egy olyan példát, amely nem létezik a közelített rendszerben, akkor az ismeretünk túlságosan absztrakt. Ekkor a

példa ellentétével, mint kiegészítő rendszerrel – ha a közelítésnek nem mond ellent – az ismeret a pontosítás irányában úgy módosítható, hogy az absztrakt lehetőségeken belül már ne tegye lehetővé a példát. Ha a példa ellentéte és a közelítés ellentmondó rendszerek, azaz nincsenek közös állapotlehetőségeik, akkor töröljük a közelítést.

Például egy alakzat-rajzoló program készítésekor észrevehetjük, hogy az alakzatok nulla szélességűre vagy hosszúságúra is méretezhetők, amely esetben természetesen eltűnik a módosított alakzat. Ezeket a nem kívánt példákat kivehetjük a programból és megadhatjuk, hogy az alakzatok nem méretezhetők egy minimális méret alá. Az alakzat-rajzolóból ugyanakkor hiányozhat az alakzatok áthelyezésének lehetősége, amellyel kiegészíthetjük a programunkat. Ugyanakkor az is egy felesleges példa lehet, hogy az alakzat a rajzolás lapján kívülre is áthelyezhető.

Tovább finomíthatjuk az ismeretszerzési folyamat módosítási lehetőségeit, ha egy kompozit rendszert szeretnénk meghatározni, mégpedig úgy, hogy a teljes rendszer módosítása mellett az egyes *komponensek változtatása* is lehetővé váljon. A σ_{+i} kiegészítő rendszerhez rendeljük egy, a lehetséges helyek halmazából vett pozíciót. A $\sigma_{+i}^{(l_i)}$ definíció szerint jelentsen olyan i -dik kiegészítő rendszert, amely csak az $l_i \in L$ helyen és az általa értelmezett időpontokban (időre vonatkozó kompozíció) módosít a közelítésen. Ez például úgy is megvalósítható, ha a $\sigma_{+i}^{(l_i)}$ olyan rendszert jelent, amelynek az általa nem, de a σ_i által értelmezett helyen és időpontokban az állapotkombinációi a σ_i megfelelő állapotkombinációival egyeznek meg.

Az ismeretszerzés során egy *közelítés* a következő módon *pontosítható*. Egyrészt a teljes rendszert vagy annak bizonyos komponenseit egy pontosabb változatra cserélhetjük. Másrészt egy független kiegészítéssel történő konjunkcióval a metszetre szűkíthetjük a felvehető állapotokat. Így a rendszert például több típus metszeteként (többszörös származtatásaként) adhatjuk meg, vagy az absztrakt lehetőségekből törölhetünk néhány példát, ha a kiegészítés a példák ellentéte. Harmadik lehetőségként egy eredetileg egységes (absztrakt) változású részt, további részletekre bonthatunk. A felbontás során először az összes részlet mind a tartalmazó résszel azonos absztrakt

állapotú, majd az egyes részleteknek pontosíthatjuk a makróállapoton belüli alállapotait.

A közelítés pontosabbá tételének pontosítás és részekre bontás változatai is megadhatók az elemi műveletekkel. Sőt, segítségükkel egy adott makróállapotú részen belüli egy vagy több részlet pontosított állapotait is leírhatjuk. Például az $\{x \in \mathbb{R} \mid x > 0\}^{(a)}$ (az a rész pozitív valós számokat vehet fel) állapotnak az $\{x \in \mathbb{R} \mid x > 0\}^{(a)} \wedge \{x \in \mathbb{R} \mid x \neq 1\}^{(ab)}$ (az a rész pozitív valós számokat vehet fel, de az azon belüli b részlet nem lehet 1) állapot a pontosítása.

▼ **Példákon keresztül történő ismeretszerzés.** A közelítésnek a még hiányzó, illetve a felesleges példák keresésén alapuló ismételt módosításával pontosabb ismereteket állíthatunk elő, így ez egy ismeretszerzés, egy tanulási folyamat. Ezt, a hiányzó, illetve felesleges példák keresésén alapuló tanulási módszert *példákon keresztül történő ismeretszerzésnek* nevezzük. Egy nem, vagy egy feleslegesen tartalmazott példa meghatározása alkalmas a megismerendő rendszer és az ismeret összehasonlítására.

Az ismeretszerzés általában valamely ábrázolási módszerrel is összekapcsolódik. A példákon keresztül történő ismeretszerzés lehetővé teszi, hogy a példának az ábrázolási módszer által kapott ábrázolásait felhasználva, valamint az absztrakciós és pontosító kiegészítések, mint módszerek, azaz függvények segítségével összetett ábrázolásokat adjunk meg. Így, ha az ábrázolási módszer rendelkezik az absztrakciós és pontosító kiegészítés jelölésével, akkor a rendszert egyszerűbb építőelemek együtteseként ábrázolhatjuk, azaz „ismert” alapelemekből „ismert” módon állíthatjuk össze.

A közelítéshez hozzáadott lehetséges példák bizonyos csoportjánál észrevehetjük, hogy azok egy típus összes lehetőségeit írják le. Ekkor az egyedi megadások egyetlen rendszerrel, a példák a típussal helyettesíthetők. Ez természetesen az ismeret ábrázolását is egyszerűsíti – annyira, hogy ha a példák nem is teszik lehetővé a típus minden lehetőségét, akkor is célszerűbb az ismeretet egy típussal és a nem lehetséges példák kizárásával megadni (pl.: $x \in \mathbb{R} \mid x \neq 0$). Korábbi definícióinkban az „absztrakció” és az „általánosítás” szavakat szinonimaként használtuk. A hétköznapi nyelvben azonban az *absztrakció* fogalma pontosan a fent leírt folyamatot, a hasonló jellegű konkrét példákból a közös lényeg kiemelését jelenti.

Ha találunk egy, a közelítésünkéből még hiányzó lefutási irányt, egy példát, akkor feltételezhetjük, hogy ez a jellegzetesség nem csak konkrét példaként, hanem típusként is jelen van a rendszerben. Az ismeretszerzést így felgyorsíthatjuk, ha a közelítést (absztrakciós irányban) nem csak a példával, hanem a példa megfelelő típusával egészítjük ki. A típus „megfelelő” voltának ellenőrzése természetesen újabb tesztelést-módosítást, azaz újabb ismeretszerzési folyamatot jelent, hogy az például ne mondjon ellent a keresett rendszernek. A hétköznapi nyelvben az *általánosítás* fogalma a példa általánosabbá, például típussá történő – esetleg ellenőrizetlen, ezért valótlan és túlzó – helyettesítését jelenti.

Analízis és szintézis

Analízisnek, azaz elemzésnek nevezzük, amikor a vizsgálatunk tárgyának, például egy rendszernek a belső szerkezetét egyszerűbb építőelemekből állítjuk össze, mely összeállítás általában a viszonyaikat jeleníti meg. Az analízis szó általában vizsgálatot jelent, amely így valamely ismeret előállítását célozza meg. *Általában* az analízis ezért valamely információ (azaz nem feltétlenül egzakt, hanem csak közelítő ismeret) meghatározását jelenti, illetve azt érthetjük a vizsgált elem és valamely más elem vagy elemek viszonyára is. *Szemléletünkben* a rendszer a teljességet jelenti, a következőkben a kiválasztott elem és más elemek viszonyának analízisét ezért az összes érintett elem együttes analíziseként tekintjük. Hasonlóan, az analízis műveletet sem a köznap értelemezés kiindulási rendszerére, hanem a megfogalmazott információra értjük. Az analízis fogalmat így olyan értelemben használjuk, hogy egy nagyobb egységet az egység adott kiindulási elemeinek adott módú összeállításaként határozunk meg. A kiindulási elemek adott összeállítását az elemek szintézisének nevezzük.

Például egy készítő információ rendszer strukturális fogalmait és viszonyait osztálydiagramokkal is megadhatjuk. Ez a megadás az ábrázolt rendszer analízise. Az implementáció során a programot a programelemek szintéziseként állítjuk elő.

Rendszert az időpontokban felvett állapotkombinációkkal, illetve kompozit esetén az adott időpontokban és helyeken felvett állapotkombinációkkal, azaz a lehetséges állapotok halmazával adunk meg. A rendszer folyamata vonatkozóan „ismertnek” nevezhetjük például azt az alkotóelemet, ame-

lyet úgy kapunk, hogy leválasztjuk annak egy komponensét és azt vagy tovább korlátozzuk, vagy annak egy hasonlítható absztrakcióját adjuk meg. Ekkor az időpontokban és helyeken felvett állapotkombinációk halmazát bővítjük vagy szűkítjük. Ugyancsak „ismertnek” nevezhetjük, ha ezt az alkotóelemet áthelyezzük egy másik időpontba vagy helyekre. Nem tekintjük azonban ismertnek azt a hasonlítható rendszert, amely újabb helyeket vagy időpontokat is értelmez, esetleg adott időpontokra és helyekre olyan állapotkombinációt állít elő, amely nem függő a többi állapotkombinációtól.

✓ **Pozícionálás, közvetlen pozícionálás.** Egy $\alpha_{\equiv} : \Sigma^{(\sigma)} \rightarrow \Sigma^{(\sigma')}$ átjelölést *pozícionálásnak* nevezünk, ha az mindössze a rendszerek időpontjait és helyeit jelöli át (így $\sigma = (\mathcal{T}, L, \overline{\mathcal{T} \times L}, S, P)$ és $\sigma' = (\mathcal{T}', L', \overline{\mathcal{T}' \times L'}, S, P')$ esetén megadható egy $\alpha_{\equiv_{\mathcal{T}}} : \mathcal{T} \rightarrow \mathcal{T}'$ (szigorúan) monoton növekvő és egy $\alpha_{\equiv_L} : L \rightarrow L'$ bijektív függvény-párral). A pozícionálást *közvetlen pozícionálásnak* nevezük, ha azonos halmazba képez ($\alpha_{\equiv} : \Sigma^{(\sigma)} \rightarrow \Sigma^{(\sigma')}$).

✓ **Tényleges és lehetséges fogalom.** Rendszer korlátozásának hasonlítható absztrakciója (korlátozásának következménye) a *rendszer tényleges fogalma*. Rendszer tényleges fogalmának közvetlen pozícionálása a *rendszer lehetséges fogalma*.

☞ *Rendszer tényleges és lehetséges fogalmai a rendszerrel hasonlíthatók.*

☞ *Rendszer tényleges fogalma egyben a lehetséges fogalma is.*

☞ *Rendszer hasonlítható absztrakciója a rendszer tényleges fogalma, mivel a korlátozás ekkor identikus.*

☞ *Rendszer példája a rendszer tényleges fogalma.*

☞ *Kompozit része a rendszer tényleges fogalma, mivel az identikus korlátozást követő hasonlítható absztrakció ekkor csak adott időpont-hely pároktól tekint el.*

✓ **Rendszer fogalma, fogalom alkalmazása.** Egy rendszer a σ rendszer fogalma (az $\alpha_{\equiv}$ rögzítése esetén), ha azonosnak tekinthető a σ egy tényleges fogalmának a megfelelő pozícionálásával. A fogalom pozícionálását a *fogalom alkalmazásának* is nevezzük.

☞ *Rendszer tényleges és lehetséges fogalmai egyben a rendszer fogalmai is.*

☞ *Rendszer fogalmai a rendszerrel hasonlíthatók.*

Rendszer fogalma esetén megengedjük, hogy mindössze egy azzal azonosnak tekintett rendszer egyezzen meg az eredeti rendszer tényleges vagy lehetséges fogalmával.

A rendszer működésében más időpontban, illetve más helyeken megjelenő ismétlődések kiemelhetők a rendszer egy lehetséges fogalmává, mely pozicionálásával megadhatjuk az időpontok és helyek tényleges megfeleltetését. A lehetséges fogalom időpontjait és helyeit átjelölhetjük „absztraktabb” időpontokká és helyekké, így a rendszer egy fogalmát kapjuk, mely esetén már elszakadhatunk az eredeti rendszer tényleges időpontjainak és helyeinek elemeitől.

✓ **Fogalom tényleges és lehetséges tartománya.** A σ rendszer egy fogalmának a *tényleges tartománya* a rendszer azon tényleges fogalmainak uniója, melyek megfelelő pozicionálásaival a fogalommal azonosnak tekinthető rendszert kapunk. A rendszer egy fogalmának a *lehetséges tartománya* a rendszer azon lehetséges fogalmainak uniója, melyek megfelelő pozicionálásaival a fogalommal azonosnak tekinthető rendszert kapunk.

☞ *Rendszer adott fogalmának lehetséges tartománya a tényleges tartományának következménye.*

A rendszer „tényleges fogalma” a rendszerre vonatkozó információnál bővebb fogalom. Egy rendszer egy nem triviális példája nem a rendszer következménye, így nem is a rendszerre vonatkozó információ, de a rendszer tényleges fogalma.

☞ *Példány tényleges fogalmai legfeljebb a példány hasonlítható absztrakciói* (mivel a példány nem korlátozható tovább), azaz a tetszőlegesség (X) kivételével mind a példányra vonatkozó információk.

Hogyan lehet tehát egy rendszer analízisét megadni? A felbontás részeként kiindulásként felhasznált alkotóelemeket megadhatjuk a rendszer fogalmaiként. Az összeépítésnek az alkotóelemek fogalmaiból egy, a rendszerrel,

így a fogalmakkal is hasonlítható rendszert kell megadni, ami valójában egy módszer, azaz egy leképezés.

✓ **Makrorendszer, építő- vagy alkotóelem.** Ha a σ rendszer előáll egy rendszer $\sigma_{i \in I}$ fogalmain alkalmazott $m^{(\hat{R})}$ módszer $\sigma = m_{i \in I}^{(\hat{R})} \sigma_i$ eredményeként, akkor az $m^{(\hat{R})}$ formát *makrorendszernek*, illetve a σ *makrorendszerként való megadásának*, valamint a σ_i fogalmakat a makrorendszer *építő- vagy alkotóelemeinek* nevezzük.

A módszer természetesen több, elemibb módszerből felépített összetett módszer is lehet.

✓ **Analízis és szintézis.** Rendszer makrorendszerként történő megadása a kiindulási rendszer *analízise* és egyben az alkotóelemek *szintézise*.

Absztrakciós és pontosító kiegészítés

Az unió és a metszet műveleteivel „ismert” rendszerekből (például egy rendszer fogalmaiból) rendkívül egyszerű módon építhetünk nagyobb, összetettebb rendszereket, azaz: makrorendszereket. Az építkezés egy lehetséges módja, ha a makrorendszer módszere egy olyan összetett függvény, amely minden egyes elemi lépésben a korábbi lépések eredménye és egy kiegészítő rendszer alapján határoz meg egy újabb részeredményt (pl. $m_1(\sigma_0, \sigma_1)$, $m_2(m_1(\sigma_0, \sigma_1), \sigma_2)$). Ebben az esetben a makrorendszer összetett függvényét két-argumentumú módszerek láncából építjük fel, melyben minden lépés az addigi részeredményen módosít.

Egy ismeret módosításánál kiköthetjük, hogy vagy csak bővítjük vagy csak szűkítjük az ismeret egyes időpontokban felvett állapotkombinációit. A bővítés az absztrakciós irányban, a szűkítés a pontosítás irányában történő kiegészítésnek felel meg.

✓ **Rendszer pontosító kiegészítése.** A $\sigma \Leftrightarrow (T, L, \overline{T \times L}, S, P)$ rendszer a kiindulási $\sigma_a \Leftrightarrow (T, L, \overline{T \times L^a}, S, P_a)$ rendszernek az azzal nem

ellentmondó $\sigma_+ \Leftrightarrow (\mathcal{T}, L, \overline{\mathcal{T} \times L}^+, S, P_+)$ rendszerrel vett pontosító kiegészítése (jelölés: $\sigma = (\sigma_a) \wedge \sigma_+$), ha $\forall (t, l) \in \mathcal{T} \times L$

$$P(t, l) = \begin{cases} P_a(t, l) \cap P_+(t, l) & \text{ha } (t, l) \in \overline{\mathcal{T} \times L}^+ \\ P_a(t, l) & \text{egyébként} \end{cases}.$$

☞ A pontosító kiegészítés a kiindulási rendszer és a kiegészítés konjunkciója, egyben azok feltétele.

☞ Az eredményrendszer $\mathcal{T}$ és S halmazának rögzítése mellett adott rendszerek konjunkciója, így egy rendszer adott, nem ellentmondó rendszerrel történő pontosító kiegészítése egyértelmű transzformáció.

☞ A pontosító kiegészítés módszer.

✓ Rendszer absztrakciós kiegészítése. A $\sigma \Leftrightarrow (\mathcal{T}, L, \overline{\mathcal{T} \times L}, S, P)$ rendszer a kiindulási $\sigma_a \Leftrightarrow (\mathcal{T}, L, \overline{\mathcal{T} \times L}^a, S, P_a)$ rendszernek a $\sigma_+ \Leftrightarrow (\mathcal{T}, L, \overline{\mathcal{T} \times L}^+, S, P_+)$ rendszerrel vett absztrakciós kiegészítése (jelölés: $\sigma = (\sigma_a) \vee \sigma_+$), ha $\overline{\mathcal{T} \times L}^+ \subseteq \overline{\mathcal{T} \times L}^a$ és $\forall (t, l) \in \mathcal{T} \times L$

$$P(t, l) = \begin{cases} P_a(t, l) \cup P_+(t, l) & \text{ha } (t, l) \in \overline{\mathcal{T} \times L}^+ \\ P_a(t, l) & \text{egyébként} \end{cases}.$$

☞ Az absztrakciós kiegészítés a kiindulási rendszer hasonlítható absztrakciója (következménye), de nem feltétlenül a kiegészítő rendszer következménye.

☞ Rendszer adott rendszerrel történő absztrakciós kiegészítései azonosnak tekintett rendszerek, így az eredményrendszer $\mathcal{T}$ és S halmazának rögzítése mellett az absztrakciós kiegészítés egyértelmű transzformáció.

☞ Az absztrakciós kiegészítés módszer.

Rendszer absztrakciós és pontosító kiegészítései esetén kizárólag a kiegészítő rendszer által értelmezett időpont-hely párok esetén módosul a kiindulási rendszer.

▽ Kompozit pontosító és absztrakciós kiegészítése. A

$\sigma = \otimes_{i \in I}^{\pi: I \rightarrow L} \sigma_i$ kompozit a kiindulási σ_a rendszer σ_+ rendszerrel vett, $l \in L$ részen történő *pontosító, illetve absztrakciós kiegészítése* (jelölés: $\sigma = (\sigma_a) \wedge \sigma_+^{(l)}$, illetve $\sigma = (\sigma_a) \vee \sigma_+^{(l)}$) esetén a kiegészítést a σ_+ -al azonosnak tekintett olyan rendszerrel végezzük el, mely a σ -val közvetlenül hasonlítható, az l része σ_+ , a többi része pedig *tetszőleges állapotú (!)*, azaz a kiegészítést a σ_+ rendszer közvetlen pozicionálásának eredményeként kapott $\sigma_+^{(l)}$ rendszerrel hajtjuk végre. Ezért a kiegészítés csak a σ kompozit l részét egészíti ki a pontosítás, illetve az absztrakció irányában, a többi részt változatlanul hagyja.

Korábban, egy kompozit ismeret absztrakciós, illetve pontosító irányba történő módosításakor a kiegészítésnek a nem módosítandó időpontokban, illetve helyeken felvett állapotkombinációjának meg kellett egyeznie a módosítandó rendszer állapotkombinációjával. Absztrakciós és pontosító kiegészítés esetén azonban a kiegészítés ezen időpont-hely párokon tetszőleges állapotkombinációjú is lehet, azaz a nem módosítandó időpont-hely párokat nem kell értelmeznie.

A lehetséges és tényleges rész

Megállapítottuk, hogy egy kiindulásként vett ismeretnek (a „teszt” összehasonlítása alapján) az absztrakció, illetve a pontosítás irányában történő kiegészítésével a közelítésünk pontosítható. Adott rész adott időpontjaiban felvehető állapotkombinációk bővítésén, illetve szűkítésén alapuló makrorendszer-képzés pontosan a közelítések lépéseinek folyamatát követi. Ezért ez a módszer *adott rendszer egy közelítését az ismeretszerzés folyamatának rögzítésével együtt ad meg*. Ellentétes irányból tekintve: **egy makrorendszer az ismeret mellett a tanulási folyamatot is megadja, azaz az ismertnek feltételezett fogalmak alapján „megismertet” az ismerettel.**

Adott ismeretszerzési folyamat kerülőútjai, illetve adott lépéseknek a későbbiekben absztrahált partikuláris eredményei átrendezhetőek egy „*ideálisabb*” ismeretszerzési folyamattá. Az ezen a módon megadott makrorendszer több, konkrét tanulási folyamatot jelent, amely a lépések sorrendjében, a partikuláris ismeretekben, illetve egyes kerülőutakban különbözhetnek egy-

mástól, de az általuk leírt rendszer azonos. Ez a megadási mód ezért egy **absztrakt ismeretszerzési folyamatot** ír le, amelyben *kevesebb a kerülőút, és a nem absztrahált partikuláris ismeret, valamint amelyben bizonyos lépések felcserélhetők.*

Az egyetlen kiegészítő rendszerrel történő pontosító vagy absztrakciós kiegészítések összevonásához szükséges, hogy ezeket a műveleteket tetszőleges számú rendszerre általánosítsuk. A kiindulási rendszerek pontosító (a felvehető állapotok szűkítésének megfelelő) irányban való összekapcsolásaként olyan műveletet kapunk, amely azonos a korábban definiált konjunkció fogalmával. A kiindulási rendszerek absztrakciós (a felvehető állapotok bővítésének megfelelő) irányban történő összekapcsolásának műveleteként definiáljuk a rendszerek diszjunkcióját.

▼ **Rendszerek konjunkciója (ekvivalens definíció).** A $\sigma \Leftrightarrow (\mathcal{T}, L, \overline{\mathcal{T} \times L}, S, P)$ rendszer a $\sigma_{i \in I} \Leftrightarrow (\mathcal{T}, L, \overline{\mathcal{T} \times L}^i, S, P_i)$ nem ellentmondó rendszerek konjunkciója (jelölés: $\sigma = \wedge_{i \in I} \sigma_i$), ha $\forall (t, l) \in \mathcal{T} \times L$ esetén

$$P(t, l) = \begin{cases} \bigcap_{i \in I \wedge (t, l) \in \overline{\mathcal{T} \times L}^i} P_i(t, l) & \text{ha } \exists i \in I : (t, l) \in \overline{\mathcal{T} \times L}^i \\ S_l & \text{egyébként} \end{cases}.$$

A konjunkció előző definíciója ekvivalens a korábbi definícióval. Ezért a konjunkció a kiindulási rendszerek többszörös származtatása, azok feltétele, valamint az eredményrendszer $\mathcal{T}$ és S halmazának rögzítése mellett adott rendszerek konjunkciója egyértelmű transzformáció.

☞ *A konjunkció egy módszer.*

§ Rendszer adott kiegészítő rendszerekkel történő, adott sorrendben vett pontosító kiegészítésének eredménye azonosnak tekinthető a rendszer és a kiegészítések konjunkciójával, **azaz a konjunkcióval a pontosító kiegészítések sorrendjétől eltekinthetünk.**

▼ **Rendszerek diszjunkciója.** A $\sigma \Leftrightarrow (\mathcal{T}, L, \overline{\mathcal{T} \times L}, S, P)$ rendszer a $\sigma_{i \in I} \Leftrightarrow (\mathcal{T}, L, \overline{\mathcal{T} \times L}^i, S, P_i)$ rendszerek diszjunkciója (jelölés: $\sigma = \vee_{i \in I} \sigma_i$), ha $\forall (t, l) \in \mathcal{T} \times L$ esetén

$$P(t, l) = \begin{cases} \bigcup_{i \in I \wedge (t, l) \in \overline{T \times L}^i} P_i(t, l) & \text{ha } \exists i \in I : (t, l) \in \overline{T \times L}^i \\ S_l & \text{egyébként} \end{cases}.$$

☞ Adott rendszerek diszjunkciói azonosnak tekintett rendszerek, így az eredményrendszer $\mathcal{T}$ és $\mathcal{S}$ halmazának rögzítése mellett a *rendszerek diszjunkciója egyértelmű transzformáció*.

☞ A diszjunkció egy módszer.

☞ A diszjunkció a kiindulási rendszerek nemdeterminisztikus absztrakciója, azok feltétele, egyben absztrakciója.

§ Rendszer adott kiegészítő rendszerekkel történő, adott sorrendben vett absztrakciós kiegészítésének eredménye azonosnak tekinthető a rendszer és a kiegészítések diszjunkciójával, **azaz a diszjunkcióval az absztrakciós kiegészítések sorrendjétől eltekinthetünk**.

A konjunkció és a diszjunkció több, eltérő sorrendű pontosító és absztrakciós lépés eredményeként kapott közelítést azonos módon jelöl. Műveleteink a halmazelmélet metszet és unió műveleteivel vannak definiálva, ezért azoknak az (asszociatív, kommutatív, stb.) tulajdonságaival is rendelkeznek. A tulajdonságok ismeretében bizonyos lépések *összevonhatók*, melyek eredményeként egy „ideálisabb”, azaz kerülőutaktól és partikuláris eredményektől mentesebb ismeretszerzési folyamatot kapunk. Az összevonás egy lehetséges eredménye a konjunkciók diszjunkciójaként kapott **diszjunkatív normálforma**, vagy a diszjunkciók konjunkciójaként kapott **konjunkatív normálforma**.

▽ **Lehetséges vagy közvetett és tényleges vagy közvetlen rész.** Rendszer adott részének hasonlítható pontosítását (a rész feltételét) a rendszer által *lehetséges részként*, *közvetve* tartalmazott rendszernek, a rész hasonlítható absztrakcióját (a rész és egyben a teljes rendszer következményét) a rendszer által *tényleges részként*, *közvetlenül* tartalmazott rendszernek nevezzük.

☞ *Rendszer által lehetséges és tényleges részként tartalmazott rendszerek a rendszer fogalmi.*

☞ *Rendszer komponense egyszerre a rendszer lehetséges és tényleges része.*

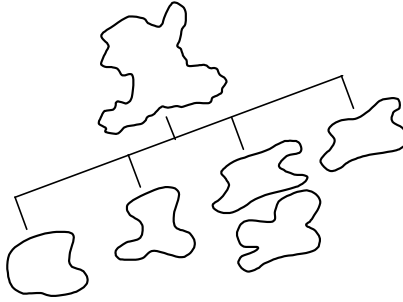
§ **Kiindulási rendszerek a diszjunkció lehetséges részei.** Egyetlen (tetszőleges argumentum-számú) diszjunkcióval megadott makrorendszer építőelemei az eredmény lehetséges részei.

§ **Kiindulási rendszerek a konjunkció tényleges részei.** Egyetlen (tetszőleges argumentum-számú) konjunkcióval megadott makrorendszer építőelemei az eredmény tényleges részei.

Adott kiindulási ismeretekből egy összetett rendszert például a következő módon képezhetünk: minden egyes kiindulási rendszer esetén adjuk meg, hogy az a makrorendszer mely részletét fogja módosítani, illetve azt, hogy értelmezett időpontjainak állapotkombinációi bővítik, vagy szűkítik a részlet lehetséges állapotait. A bővítések absztrakciós ($\vee$), a szűkítések pontosító ($\wedge$) kiegészítéseként adhatók meg. Az egymást követő módosítások összevonhatók. A makrorendszerek az alkotóelemek és módosító irányuk sorrendtől független, azaz tetszőleges sorrendű megadását úgy kapjuk, hogy az azonos helyre és időre vonatkozó kiegészítéseket az *összevonás* során egyetlen kiegészítéssel helyettesítjük. A kiegészítések összevonása és egyszerűsítése után az eredményül kapott makrorendszer bővítés esetén „vagy” jellegű közvetett lehetőségként, szűkítés esetén pedig „és” jellegű közvetlen ténylegességként, például részként vagy a rendszer korlátozójaként tartalmazza az alkotóelemet, a hasonlíthatóvá átjelölt kiegészítést. Az eszközzel definiálható egy speciális makrorendszer, a *konstrukció*.

▽ **Rendszerek konstrukciója.** Az építőelemekhez rendeljük egy-egy pozíciót, valamint egy-egy jelzőt, hogy az elem bővíti vagy szűkíti az általa értelmezett időpontokban a pozícióhoz tartozó rész állapotkombinációit. A kapott makrorendszert az alkotóelemek *konstrukciójának* nevezzük.

A konstrukció egy speciális fajtáját, az *egyszerű konstrukciót* kapjuk, ha az eszközkészletet tovább egyszerűsítjük. A kiindulási rendszerek átjelöléséhez, az alkotóelemekhez mindössze egy-egy pozíciót rendelünk és azokat a pozícióknak megfelelő részek egy-egy lehetséges példájaként értelmezzük. Az eredmény így az azonos részeket meghatározó elemek diszjunkcióinak kompozíciója lesz.



19. ábra. Rendszerek egyszerű konstrukciója.

▽ **Egyszerű vagy környezetfüggetlen konstrukció.** Az alkotóelemekhez egy-egy pozíciót rendelve az elemeket egyszerű kiegészítéseként, azaz a pozíció által meghatározott rész egy-egy lehetséges példajaként értelmezzük. A kapott konstrukciót az alkotóelemek *egyszerű vagy környezetfüggetlen konstrukciójának* nevezzük.

Az egyszerű konstrukción alapuló technikával megadható a „kiegészítés” olyan módszere, amely megfelelő kompozícióival egy közelítés az absztrakciós és a pontosítás irányába is módosítható.

▽ **Rendszer egyszerű kiegészítése.** A $\sigma \Leftrightarrow (\mathcal{T}, L, \overline{\mathcal{T} \times L}, S, P)$ rendszer az a $\sigma_a \Leftrightarrow (\mathcal{T}, L, \overline{\mathcal{T} \times L}^a, S, P_a)$ rendszer $\sigma_+ \Leftrightarrow (\mathcal{T}, L, \overline{\mathcal{T} \times L}^+, S, P_+)$ rendszerrel vett egyszerű kiegészítése (jelölés: $\sigma = (\sigma_a) \diamond \sigma_+$), ha $\forall (t, l) \in \mathcal{T} \times L$ esetén

$$P(t, l) = \begin{cases} P_+(t, l) & \text{ha } (t, l) \notin \overline{\mathcal{T} \times L}^a \wedge (t, l) \in \overline{\mathcal{T} \times L}^+ \\ P_a(t, l) \cup P_+(t, l) & \text{ha } (t, l) \in \overline{\mathcal{T} \times L}^a \wedge (t, l) \in \overline{\mathcal{T} \times L}^+ \\ P_a(t, l) & \text{egyébként (ha } (t, l) \notin \overline{\mathcal{T} \times L}^+) \end{cases} .$$

Az egyszerű kiegészítés a σ_+ által értelmezett, de σ_a által nem értelmezett időpontokban a felvehető állapotok szűkítése, míg a σ_+ és σ_a által egyaránt értelmezett időpontokban az állapotok bővítése. A σ_+ által nem

értelmezett időpillanatokban az állapotkombinációk változatlanul maradnak. Az egyszerű kiegészítéssel így a felvehető állapotok bővítése és szűkítése is kifejezhető. A megfelelő állapotkombinációval történő bővítéssel ugyancsak kifejezhető egy már értelmezett időpont megadott állapotának törlése is.

☞ *Rendszer adott rendszerrel történő egyszerű kiegészítései az eredményrendszer T és S halmazának rögzítése mellett az egyszerű kiegészítés egyértelmű transzformáció.*

☞ *Az egyszerű kiegészítés módszer.*

Bár az eszközkészletünk egyszerűsödhet, az egyszerű kiegészítésekből alkotott sorozat (összetett módszer) nagyon nehezen rendezhető át, mivel például az egyszerű kiegészítések nem asszociatívak. (Pl. $S=\{1,2,3\}$ esetén egy $P(t)=\{1,2\}$ állapotkombináció $\{3\}$, majd $\{2\}$ kombinációkkal történő kiegészítése $\{2\}$ kombinációt, átcsoportosítva pedig $\{1,2,3\}$ kombinációt eredményez.) A sorrendtől független, „tisztá” változatnak az egyszerű konstrukció tekinthető.

III

Racionalitás és analógia

A hétköznapijainkban is gyakran használt modell fogalmát egy speciális ábrázolásként definiáljuk, amely szemantikája, az ún. „absztrakt modell”, függvény-jellegű kapcsolatban van az eredeti rendszerrel. A rendszerek, így az ismeretszerzések egyszerűen ábrázolhatók, ha a változást csak egymásutáni „mértőldköveknel”, azaz lépésekben vizsgáljuk. A lépésenkénti véges választásokon alapuló ismeretszerzések folyamatát követő ábrázolások elvezetnek a kiindulási rendszer véges, racionális és valós modelljeihez.

Modell

A modell fogalmával gyakran találkozunk mind hétköznapijainkban, mind a rendszerszervezések és fejlesztések során. Láthattunk hajó- és repülőmodelleket, kémiai molekulák modelljeit, stb.

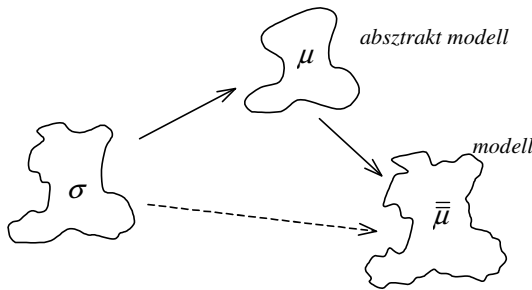
A rendszerszervezési technológiákban, így például az objektumorientált módszerekben is központi szerepet kap a modell fogalma. Magában az OMT „Objektum Modellezési Technika”, valamint az UML Unified Modeling Language elnevezésében is megjelenik, illetve fő célkitűzésükben, azaz a „valóság objektumainak modellezésében”. Magára a modell fogalmára azonban csak egy rendkívül vázlatos definíciót kapunk. Az OMT megfogalmazása szerint: „A modell az valaminek az absztrakciója, hogy megérthessük azt, annak elkészítése előtt”. Ez a mondat a modellt egy absztrakcióként határozza meg.

Szemléletünk definíciói szerint maga a modell nem az eredeti „valami” absztrakciója; az absztrakciós viszony csak a modell jelentésére, szemantikájára igaz. Így a következőkben megkülönböztetjük a jelentést tisztán tartalmazó absztrakt modellt és annak megjelenési formáját, a modellt.

Egy hajómodell mögött álló absztrakt modell az eredeti hajó absztrakciója lesz. Értelmezésünkben a modell tehát egy ábrázolás, mégpedig egyszerű

ábrázolás. Az absztrakt modellre pedig további kikötést is teszünk: a hétköznapi értelmezéshez közeli módon a modell jelentésének, azaz az absztrakt modellnek a kiindulási rendszer típusabsztrakciójának kell lenni, azaz a rendszer és absztrakt modellje között egy függvényjellegű (!) kapcsolatot határoz meg.

▼ **Absztrakt modell és modell.** A σ rendszer μ típusabsztrakciója a σ *absztrakt modellje* (a σ absztrakt modelljeinek jelölése: $\mathcal{M}\sigma$). A μ rendszer (vele hasonlítható) $\bar{\mu}$ példánya a σ rendszer *modellje* (a σ modelljeinek jelölése: $\bar{\mathcal{M}}\sigma$).



20. ábra: Absztrakt modell és modell.

Az absztrakt modellnél tágabb fogalom az absztraktum. Ha egy kiindulási rendszert további példákkal egészítünk ki, azaz végrehajtunk egy absztrakciót, akkor eredményül nem feltétlenül a kiindulási rendszer típusabsztrakcióját, azaz absztrakt modelljét kapjuk. Ebben az esetben megadható egy olyan további absztrakció, amely egy olyan absztrakt modellt eredményez, ami egyben a kiindulási rendszerek absztrakt modellje is lesz.

Példa: Modell és absztrakt modell.

Egy ember egy oszlopot – mellette elhaladva – megfog és ahhoz a karizma megfeszítésével egyre közelebb kerül, így a sebessége is nagyobb lesz. Ezzel megalkotta a bolygó mellett elhaladva felgyorsító mesterséges hold modelljét. A modell mögötti lényeg – a szemantika – az absztrakt modell, amely a fizika törvényeivel írja le a centripetális gyorsuláshoz szükségesnél nagyobb erővel körmozgásra kényszerített test kerületi sebességének növekedését.

§ **Modell és egyszerű ábrázolás.** Rendszer absztrakt modellje a kiindulási rendszer absztrakt ábrázolása (mivel a típusabsztrakció átírható egy hasonlítható típusabsztrakcióvá és egy átjelöléssé), modellje pedig az egyszerű ábrázolása. Absztrakt modell esetén az $A_{(\alpha)}\sigma$ absztrakt ábrázolás, illetve modell esetén az $A_{(\alpha')}^{\sigma}$ egyszerű ábrázolás (egyben absztrakt átjelölés) α absztrakcióitól megköveteljük, hogy azok típusabsztrakciós módszerek legyenek.

§ **Absztrakt modell és modell pontossága.** Az absztrakt modell egy közelítő vagy egzakt ábrázolás, ezért a pontossága önmaga. Modell pontossága pedig a modell absztrakt modellje.

Rendszer absztrakt modellje és modellje a rendszer ábrázolása, ezért beszélhetünk *egzakt*, illetve *közelítő* absztrakt modellről és modellről.

▽ **Egzakt és közelítő absztrakt modell és modell.** A σ rendszerrel ekvivalens μ a σ típusabsztrakciója, így absztrakt modellje, egyben egzakt ábrázolása, ezért azt *egzakt absztrakt modellnek* nevezzük. Egzakt absztrakt modell példánya a kiindulási rendszer *egzakt modellje*. Ha a σ rendszer μ absztrakt modellje nem ekvivalens a σ rendszerrel, akkor μ -t a σ *közelítő absztrakt modelljének*, a μ példányát pedig a σ *közelítő modelljének* nevezzük.

☞ *Egy tetszőleges absztrakt rendszer bármely rendszer absztrakt modellje.*

☞ *Egy rendszer bármely példánya egyben önmaga (egzakt) modellje is.* Ugyanakkor adott rendszer tetszőleges példánya tetszőleges rendszer modellje, mivel a kettő egy tetszőleges absztrakt rendszert eredményező típusabsztrakcióval összekapcsolható.

A típusabsztrakció egy „jól viselkedő” absztrakció, mivel a rendszer és típusa között egy függvény-jellegű kapcsolatot ír le: a kiindulási rendszer időpontjainak kiválasztása és átjelölése mellett az állapotok osztályokba csoportosítását és átjelölését adja meg. Ezért a modellként történő ábrázolás is egy „jól viselkedő” ábrázolás, mivel a kiválasztott időpontokban az állapotosztályok és állapotkombinációk között egy függvény-jellegű kapcsolatot ír le.

Modell pontosítására két lehetőségünk van. Egyrészt az absztrakt modell egészét vagy egyes részeit módosíthatjuk úgy, hogy egyes típusként értelmezett állapotokat típuspontosítással a típus megfelelő elemével helyettesítsünk. Ennek speciális formája, ha néhány eddig ismert időpontban egyszerű pontosítással megadjuk egy eddig bizonytalan állapotú rész absztrakt változását. Másik lehetőségként az absztrakt modellt további időpontokban is értelmezhetjük. Első esetben a modell pontosított állapotokkal, második lehetőség esetén pedig nagyobb időbeni részletezettséggel követi a változásokat.

§ Kompozíció és modell. Ha $\sigma = \otimes_{i \in I} \sigma_i$ és $\mu = \otimes_{i \in I} \mu_i$ kompozitok esetén minden $i \in I$ -re μ_i a σ_i típusabsztrakciója, akkor μ egyben σ típusabsztrakciója is. Ekkor μ bármely $\bar{\mu}$ példánya is előáll $\bar{\mu}_i$ részek kompozíciójaként úgy, hogy minden $\bar{\mu}_i$ egyben a megfelelő μ_i példánya. Tehát minden σ_i résznek a megfelelő μ_i az absztrakt modellje, $\bar{\mu}_i$ pedig a modellje, egyben σ rendszernek μ az absztrakt modellje, $\bar{\mu}$ pedig a modellje.

Kvantumrendszer és kvantált rendszer

Általános rendszer esetén az időpontok halmaza egy tetszőleges rendezett halmaz, így lehet például a valós számok folytonos halmaza is. A változás függvényének minden időponthoz meg kell adni a rendszer által felvett állapotkombinációt.

Rendkívül könnyen kezelhetők, például könnyen ábrázolhatók azok a rendszerek, amelyekben minden értelmezett időpont után meg tudjuk határozni a következő értelmezett időpontot, ha valóban létezik későbbi időpont. Például, egy rendszer változásának vizsgálata során az éppen vizsgált időpont esetén vagy meghatározzuk a következő vizsgált időpontot („vizit egy hét múlva”) vagy befejezzük a vizsgálatot („meggyógyult”). A közbülső időtartományban pedig tetszőlegesnek vesszük a változást.

Folyamatosság esetén a változás függvényének minden közbülső értelmezett időpontra meg kell adni a felvett állapotkombinációt. A lépésenkénti változás minden lépése azonban egy-egy pillanatfelvétel-jellegű képet mutat, amelyek egymás utáni állapotkombinációkként, azok sorozataként jelennek meg. A vizsgálat kiemelt időpontjai az eredeti időhalmazt időtartamokra

bontja, minden időpont után egy időmennyiség, azaz *időkvantum* után történik meg a következő vizsgálat. Az így kapott kép-sorozatban az eredetileg folyamatos változás ugrásszerű átmenetekként, az állapotok kvantum-ugrásaiként jelenik meg.

A lépésekben történő változást ezért kvantum-változásnak, illetve kvantumrendszernek nevezzük, egy rendszer ily módon megalkotott képét pedig kvantált rendszernek.

✓ **Rendezett kvantumhalmaz, rákövetkező.** Az A rendezett halmaz *rendezett kvantumhalmaz*, ha egyetlen eleme van, vagy ha minden $a < a_g$ ($a, a_g \in A$) esetén $\exists a' \in A: a < a' \leq a_g$ (a' az a „rákövetkezője”, a az a' „előzője”), hogy nincs olyan $a'' \in A$, hogy $a < a'' < a'$. Az a rákövetkezőjét a' módon jelöljük.

☞ Egy rendezett halmaz pontosan akkor rendezett kvantumhalmaz, ha ahhoz kölcsönösen egyértelmű és szigorúan monoton módon hozzárendelhetők az egész számok egy részhalmaza.

✓ **Kvantumhalmaz.** Az A halmaz *kvantumhalmaz*, ha azon definiálható olyan rendezés, mely esetén rendezett kvantumhalmazt kapunk.

☞ Egy kvantumhalmaz legfeljebb megszámlálható számosságú.

✓ **Kvantumrendszer.** Egy rendszer *kvantumrendszer*, ha az értelmezett időpontok halmaza rendezett kvantumhalmaz.

✓ **Kvantált rendszer.** Adott kiindulási rendszer kvantált képét *kvantált rendszernek* nevezzük, ha értelmezett időpontjai a kiindulási rendszer értelmezett időpontjainak egy részhalmaza, egyben kvantumhalmaz, valamint ezekben az időpontokban a kiindulási rendszerrel azonos állapotkombinációkat vesz fel.

§ **Kvantált rendszer a kiindulási rendszer kvantált absztrakt modellje.** Rendszer kvantálása esetén a változást csak a kiválasztott pillanatokra szűkítve vizsgáljuk és eltekintünk a köztes időpillanatokban felvett állapotoktól. A kvantált rendszer ezért a rendszer hasonlítható típusabsztrakciója, azaz absztrakt modellje.

Kvantumváltozás

Kvantumrendszer változása „kvantumugrásokként”, átmenetként jelenik meg. Kis túlzással a kvantált rendszer az eredeti változás „eredményorientált” képét alkotja meg, mivel csak az előző ponthoz viszonyított változásra koncentrálnak, a közbeni folyamatoktól pedig eltekint. Nem véletlen, hogy a projekt-irányító, illetve egyes rendszerfejlesztési módszerekben a projektvezetés által alkotott kép a mérföldkövekre helyezi a hangsúlyt. A mérföldkövek lehetnek időpontok („leadási határidő”), de ugyancsak mérföldkövek egy eredmény adott időpontban történő elérése (pl. a „beta-teszt elkészülte”).

Rendszer kvantált képe leegyszerűsíti mind a változás vizsgálatát, mind annak ábrázolását.

A következőkben az átmenetekre teszünk néhány megállapítást. Megállapításaink során a megfogalmazások egyszerűsítése miatt a *továbbiakban a rendszerek determinisztikus jelölését használjuk*. Korábban láttuk, hogy ezzel nem sérül a megállapítások általánossága, mivel a determinisztikus és a nemdeterminisztikus forma egyszerűen egymásba alakítható.

Determinisztikus megadás esetén a kvantumátmenet egy állapottranszformációként jelenik meg, mivel egy adott t -ben felvett állapothoz a rákövetkező t' időpontban felvett állapot rendelődik $((t, s) \mapsto (t', s'), \text{ illetve } s \mapsto s')$. Az átmenet során a rendszer a kiinduló állapotból a következő állapotba lép.

Kvantált rendszer értelmezett időpontjai – a „mértföldkövek” – a kiindulási rendszer teljes $\mathcal{T}$ időhalmazát intervallumokra, *időkvantumokra* bontják. Két, egymás utáni mérföldkő a $\mathcal{T}$ egy intervallumának kezdő- és zárópontja. Az átmenet az intervallum kezdőpontjában felvett állapotból a zárópontban felvett állapotba vált át. A mérföldkövek az egyes intervallumok határpontjai. Minden egyes mérföldkő az előző mérföldkővel (ha az létezik) alkotott intervallum zárópontja, egyben a következő mérföldkővel (ha az létezik) alkotott intervallum kezdőpontja.

Kvantált rendszer esetén a változás az egyes intervallumok kezdőpontjában felvett állapotból a zárópontban felvett állapotba való ugrásokként jelenik meg, az intervallumon belüli változástól pedig eltekintünk. Ha a kvantálással kapott rendszer kevés információt ad a kiindulási rendszerről, akkor

egy intervallumon belüli újabb értelmezett időpontok és az ott felvett állapotok felvételével pontosíthatunk az ismereten.

Az egyetlen időpontban vizsgált rendszer a változásnak csak egy statikus pillanatfelvétele. A változás, a dinamika vizsgálatához legkevesebb egy viszonyításként választott kiindulópontban és egy későbbi eredménypontban (egy intervallum kezdő- és zárópontjában) felvett állapotok, azaz egy átmenet vizsgálata szükséges. A kvantum állapot-változás, az átmenet így a *változás kvantumaként*, a változás legkisebb vizsgálható egységeként jelenik meg.

▼ **Kvantumváltozás, állapotváltás vagy lépés.** Adott rendszernek a két értelmezett időpontjára leszűkített változása (rendszer két értelmezett időponttal) a „változás kvantuma”, a rendszer *kvantum állapotváltozása*, állapotváltása vagy *lépése*. A lépés tekinthető állapottranszformációként, amely a t értelmezett időpontban felvett s állapothoz a t' -ben felvett s' állapotot rendeli.

▼ **Kiinduló- vagy startpont, édeni állapot.** Ha egy rendszer esetén az értelmezett időpontok halmazának létezik minimuma, akkor azt *kiinduló- vagy startpontnak*, a rendszer ott felvett állapotát pedig *édeni állapotnak* nevezzük. (Az „édeni állapot” elnevezés az informatikában a sejtautomatákkal kapcsolatban jelent meg.)

§ **Startpontú kvantumrendszer változása ábrázolható az állapotok sorozataként.** Egy $\sigma = (\mathcal{T}, T, S, P)$ t_0 startpontú kvantumrendszer esetén megadható egy vele ekvivalens $\sigma = (\mathcal{T}', N', S, P')$ (N' a természetes számok halmaza, vagy annak $0 \dots n$ részhalmaza) rendszer. A természetes számokon vagy azok $0 \dots n$ részén értelmezett $P': N' \rightarrow S$ változás megadható az állapotok s_i ($i \in N'$) véges vagy végtelen sorozataként: $s_0 \mapsto s_1 \mapsto s_2 \mapsto \dots \mapsto s_n$ vagy $s_0 \mapsto s_1 \mapsto s_2 \mapsto \dots$ (s_0 utáni s_1 állapot, az utáni s_2 állapot, stb.).

☞ *Kvantumrendszer adott időpont utáni értelmezett változása megadható az állapotok sorozataként, ahol a startpont az adott $t \in \mathcal{T}$ időponttal azonos vagy az utáni első értelmezett $t < t_0 \in T$ időpont.*

A kvantumátmenet egy kvantumugrást jelent, az adott állapotból a következő állapotba történő váltást. A nem-kvantumátmenet egy olyan függvénnyel adható meg, amely egy időpontban felvett állapothoz a későbbi értelmezett időpontok függvényében határozza meg az ott felvett állapotot ($P^{(s_i)}: T' \rightarrow S$, ahol $T' \subseteq T$ a t -nél későbbi időpontok halmaza), így ez a forma folytonos átmenet leírására is alkalmas. A nem-kvantumátmenettel analóg a rendszerek (determinisztikus) megadása, amely nem egy viszonyítási pont alapján, hanem globálisan határozza meg az értelmezett időpontokban felvett állapotokat.

Ismeretszerzés mint rendszer

Egy σ rendszer $\sigma_{i \in I}$ kontrakciója önmagában is egy ismeretszerzési folyamatot ad meg, mivel előállít legalább egy, nem triviálisan pontosabb ismertetet. Korábbi megfogalmazásunk alapján a kontrakció tekinthető egy „ideális” ismeretszerzésnek, mivel monoton közelítés, azaz egyrészt a σ rendszernek csak hasonlítható absztrakcióit adja meg, azaz egyetlen ismeret sem hibás, hanem közelítő vagy egzakt ismeret. Másrészt, minden σ_i az összes korábbi ismeret hasonlítható pontosítása, ezért a kontrakció minden i -hez tartozó $p_i^{(<)}$ pontossága bármely $<$ szempont esetén maga a σ_i , azaz a kontrakció pontossága önmaga.

Egy kontrakció – mint „ideális” ismeretszerzés – több ismeretszerzési folyamat és nézőpont párok halmazát határozza meg, nevezetesen, egy $\kappa_{i \in I}$ kontrakció a $p_i^{(<)} \Leftrightarrow \kappa_i$ pontosságokkal rendelkező ismeretszerzéseket.

A kontrakció, mint „ideális” ismeretszerzés ezért alkalmas arra, hogy egy ismeretszerzést csak az eredményeinek tekinthető pontosságainak erejéig határozzon meg, azaz egy kontrakció több ismeretszerzés halmazát, egy *absztrakt ismeretszerzést* ad meg.

☞ *Ismeretszerzés pontosságai ismeretszerzést alkotnak.*

Egy $\sigma_{i \in I}$ ismeretszerzés esetén az I rendezett halmaz, ezért az ismeretszerzés maga is tekinthető rendszernek. Ha az i „időpontokhoz” állapotként az eredményeknek tekinthető, az ismeretszerzés $<$ szempontja szerinti $p_i^{(<)}$

pontosságokat rendeljük, azaz vesszük az ismeretszerzés pontosságait, akkor az így kapott rendszer absztrakt módon írja le az ismeretszerzési folyamatot, illetve a rendszer maga is egy „ideális” ismeretszerzési folyamat, egy kontrakció.

✓ **Absztrakt ismeretszerzési folyamat.** A $\sigma_{i \in I}$ absztrakt ismeretszerzési folyamat esetén az I időpontjaihoz állapotként az ismeretszerzés $\prec$ szempontja szerinti $p_i^{(\prec)}$ pontosságokat rendeljük.

☞ *Az absztrakt ismeretszerzési folyamat több ismeretszerzési folyamat halmazát határozza meg.*

§ **Az absztrakt ismeretszerzési folyamat egy kontrakció, illetve minden kontrakció egy-egy absztrakt ismeretszerzési folyamat.**

Ha egy ismeretszerzési folyamat a rendszerre vonatkozó egyre pontosabb ismereteket állít elő, akkor az ismeretszerzés és az absztrakt ismeretszerzés mint rendszer maga is rendelkezik idővel, azaz szigorúan változó rendszer lesz.

✓ **Szigorúan monoton közelítés.** A σ rendszer $\sigma_{i \in I}$ monoton közelítése a σ szigorúan monoton közelítése, ha $\forall i < j$ ($i, j \in I$) esetén σ_i és σ_j nem azonosnak tekintett rendszerek (a σ_j nem triviális feltétele).

☞ *Szigorúan monoton közelítés mint rendszer szigorúan változó.*

☞ *A σ szigorúan monoton közelítése a σ kontrakciója, ha az I legalább kételemű.*

Egy ismeretszerzési folyamat kvantálása azt jelenti, hogy a folyamatot csak bizonyos mérföldkövek esetén vizsgáljuk, a köztes állapotoktól pedig eltekintünk.

✓ **Rendszer kvantum vagy lépésenkénti közelítése.** A σ rendszer esetén a (vele hasonlítható) $\sigma_{i \in N}$ (N a természetes számok halmaza, vagy annak $0 \dots n$ részhalmaza) ismeretek sorozata a rendszer lépésenkénti közelítése.

§ **Kvantált ismeretszerzési folyamat ábrázolható sorozatként**, mely elemei az ismeretek halmaza, mivel az ismeretszerzési folyamatnak van kezdőpontja, ezért a kvantált képe egy startpontú kvantumrendszer.

§ **Kvantált absztrakt ismeretszerzési folyamat ábrázolható a pontosságok sorozatként**, mivel az absztrakt ismeretszerzési folyamat maga is (egy „ideális”) ismeretszerzési folyamat.

☞ *Kvantált ismeretszerzési folyamat egy lépésenkénti közelítés.*

Az ismeretszerzés kvantálásakor egy kiválasztott következő mérföldkő lehet például az újdonságot jelentő pontosságú új σ_i ismeret, azaz amely pontossága nem az eddigi $p_i^{(<)}$ pontosságok következménye.

§ **Kontrakció kvantálható legalább kételemű, szigorúan monoton közelítéssé, mely maga is egy „ideális” ismeretszerzési folyamat.** Ehhez mindössze olyan időpontokat kell a kontrakcióból kiválasztanunk, melyek közelítései a korábbi ismeretek szempontjából vett újdonságok, azoknak nem triviális feltételei. Kiválasztható legalább két ilyen közelítés, mivel a kiindulási rendszerünk kontrakció, ezért a kezdőpont közelítésénél létezik egy nem triviálisan pontosabb közelítés.

Egy kontrakcióként megadható absztrakt ismeretszerzési folyamat, több, azonos pontosság-változású ismeretszerzési folyamat absztrakt képe. Az absztrakt ismeretszerzési folyamat kvantálása a konkrét ismeretszerzési folyamatoknak egy nagyobb körét, bővebb halmazát határozza meg, azokat, melyeknél a vizsgált mérföldkövek pontosságai megegyeznek.

Az ismeretszerzések rendszerként, folyamatként történő megközelítésével egyszerűsíthetjük azok vizsgálatát. Ha eltekintünk a tényleges ismeretekről és csak a szempont szerinti pontosságukat vesszük figyelembe, akkor több ismeretszerzési folyamat absztrakcióját kapjuk, mely egyben egy „ideális” ismeretszerzés, így csak közelítő vagy egzakt (azaz nem hibás) ismereteket állít elő, melyek pontosságai önmaguk. Egy absztrakt ismeretszerzési folyamat kvantálásával pedig eltekinthetünk bizonyos köztes (újdonságot nem hozó sikertelen vagy „nem túlságosan nagy újdonságú”) lépésektől, így a konkrét ismeretszerzések még bővebb halmazát foghatjuk át. Kvantálással

az ideális ismeretszerzési folyamat az azonos pontosságú vagy egyre pontosabb közelítések egyszerű sorozataként írható le. A mérföldkövek megfelelő kiválasztásával elhagyhatók az újdonságot nem hozó köztes ismeretek, így ezen a módon szigorúan monoton közelítést kapunk. Ugyanakkor a kvantálással az eredeti ismeretszerzési folyamat absztrakt modelljét, egyben arra vonatkozó információt kapunk. Az ismeretszerzések rendszerként történő megközelítése, az absztrakt ismeretszerzési folyamat, illetve annak kvantálása így alkalmas arra, hogy konkrét ismeretszerzéseket „eredményorientált” módon közelítsünk meg. Elvonatkoztatva az adott tanulási folyamatok lényegtelennek tartott részleteitől csak a tényleges, illetve lényegesnek tartott közelítő vagy egzakt ismereteket emeljük ki.

Ismeretszerzés mint kiválasztás

Megközelítésünkben gyakran jelenik meg a választás mint eszköz, azaz egy halmaz adott részhalmazának a meghatározása. A rendszer definíciója maga is kiválasztásokon alapszik, mivel az egyes időpontokban a teljességet jelentő teljes állapothalmaz egy részhalmazaként adjuk meg az időpontban lehetséges állapotok halmazát. A hasonlítható pontosítás ugyancsak kiválasztásokon alapszik, mivel szűkíteni kell az egyes időpontokban felvehető állapotkombinációkat, azaz ugyancsak egyes halmazok részhalmazát kell meghatározni.

Az ismeretszerzésnek a kezdőpont ismereténél pontosabb rendszert kell meghatározni, ezért a kiválasztás az ismeretszerzés folyamatában is alapvető szerepet játszik. Egy ismeretszerzés során egy közelítő ismeretet (például egy nem egzakt ismeret pontosságát) a következőképpen pontosíthatunk. Mivel az ismeret közelítő és nem egzakt, ezért a rendszer nem triviális hasonlítható absztrakciója, így a rendszernél több lehetőséget is megenged. Ha meghatározzuk a közelítés által leírt lehetőségek lehetséges alváltozatait, akkor ki kell választanunk közülük azokat, amelyek nem hibás közelítések. Ha az összes lehetséges alváltozatot diszjunkt halmazokra bontjuk, akkor determinisztikus rendszer közelítése esetén a kiválasztás eredménye egyetlen osztály lesz, mely a kiindulási ismeretnél egy pontosabb közelítő vagy egzakt ismeret.

Választáson általában egy kvantált műveletet értünk (amely egy halmazhoz annak egy részhalmazát rendeli), így a választások egymás utáni lépé-

sekben követik egymást. A „folytonosan lezajló választást” úgy értelmezhetjük, amely minden időponthoz (vagy a választás folyamatának megfelelő koordináta adott pontjához) megadja a megfelelő, egyre kisebb elemszámú részhalmazt.

A kiválasztás módszerét tovább korlátozhatjuk, ha egy lépésben csak *véges számú lehetőség*ből történő választást engedélyezünk, amely pedig tovább bontható egyszerű igen/nem döntésekké. A lépésenkénti választások módszere egy sajátos ábrázolási módot implicál, a véges szimbólum-halmazból képzett sorozattal történő ábrázolást, amelyet *leírásnak* nevezünk. A leírás egy speciális változata a *szimbolikus modell*, amely az ábrázolt rendszer modellje, azaz egy egyszerű ábrázolása.

A szimbolikus modellek segítségével meghatározhatjuk a választásokon alapuló közelítések minőségi fokozatait. *Feltételezve* (!), hogy a tudatos megismerés, a racionalitás, a logika alapvetően véges számú lehetőségből történő *választások*, végső soron pedig egyszerű igen/nem *döntések* lépésein alapszik, eljuthatunk a tudatosságnak a természetről alkotott racionális, és annak kiterjesztéseként kapott valós világképéig. *Ez természetesen filozófia* és a hipotézisek világa. Mégis, a véges számú lehetőségből történő választások lépéseinek engedélyezett száma egyértelműen a természetes, a racionális, illetve valós számfogalomig vezet el.

Lépésenkénti véges választású közelítés

Adott rendszerre vonatkozó minden egyes információ a rendszer egy-egy közelítése. A *közelítés technikája* szükségszerűen behatárolja a rendszerről megszerezhető információk körét, egyben közvetve a megszerzett információk tárolásának, ábrázolásának módját is meghatározza. Korábban láttuk, hogy a példákon keresztül történő megismerés ábrázolható a példák konjunkciója és diszjunkciója segítségével, illetve általában, a makrorendszerként történő megadás egy ismeretszerzési folyamatot is tartalmaz, azaz „megismertet” a rendszerrel. Az ábrázolásokból így a közelítés módszerére is következtethetünk, illetve hasonlóan, az *ábrázolás technikája* ugyancsak korlátozza az ábrázolható információk körét. Az egyes ábrázolási módok így hierarchiákba rendezhetők, melyek egyes szintjei az *információ* ábrázolásának minőségi fokozatait határozzák meg.

▽ **Lépésenkénti véges választású közelítés.** Egy vizsgált *determinisztikus megadású* rendszer közelítését hajtsuk végre a következő módon! A rendszer lehetséges állapotait osztályozzuk néhány (a „néhány” jelzőt a következőkben a „véges számú” szinonimájaként használjuk) absztrakt állapotba, majd a rendszer idejéből kiválasztva egy időpontot, határozzuk meg, hogy ott a rendszer melyik osztály absztrakt állapotában van. Ezután vagy a rendszer idejéből egy újabb időpontot kiválasztva az eddigi osztályozás alapján meghatározzuk a rendszer ott felvett absztrakt állapotosztályát, vagy egy már kiválasztott időpontban meghatározott állapotot pontosítsunk úgy, hogy az absztrakt állapotosztályt további néhány (azaz véges számú) alosztályra bontjuk. Egyetlen feltételünk, hogy az absztrakt állapotok osztály-hierarchiája a teljes rendszerre egységes legyen, azaz egy osztálynak csak egyetlen alosztályokra bontását engedélyezzük. A módszer minden egyes lépése után kapott rendszer egyre pontosabban közelíti meg a vizsgált rendszert. Minden rendszer értelmezett időpontjainak és állapotainak halmazai – mivel véges számú osztályt véges lépésben véges alosztályokra bontottunk – véges halmazok. A rendszerek így kapott sorozatát ezért a vizsgált *rendszer lépésenkénti véges választású közelítésének* nevezzük. Minden lépés eredménye az előző rendszerek típuspontosítása, egyben a vizsgált rendszer típusabsztrakciója, azaz absztrakt modellje. Egy ilyen közelítő rendszer állapothalmaza az osztály-hierarchia legalsó szintjén álló alosztályokat tartalmazza. Ha egy kiválasztott időpontban nem határoztuk meg ennyire pontosan az állapotot, akkor ott – nemdeterminisztikus megadási módon – felvehető állapotként az osztály összes legalsó szintű alosztályát fel kell sorolni.

▽ **Tetszőleges pontosságú közelítés.** A σ rendszer $\sigma_{i \in I}$ kontrakciója a σ *tetszőleges pontosságú közelítése*, ha $\forall i < j$ esetén σ_i és σ_j nem azonosnak tekintett rendszerek (σ_i a σ_j nem triviális feltétele) vagy σ_j és σ azonosnak tekintett rendszerek.

A tetszőleges pontosságot abban az értelemben használjuk, hogy bármely, a kiindulási rendszerrel nem azonosnak tekintett ismeretnél minden „későbbi” ismeret (nem triviálisan) pontosabb közelítés lesz.

§ **A lépésenkénti véges választású közelítés egy tetszőleges pontosságú közelítés,** mivel a közelítés minden újabb lépése az előző lépések nem triviális (típus)pontosítása.

§ A lépésenkénti véges választású közelítés egy kontrakció, és így egyben egy ismeretszerzési folyamat.

Állapot lépésenkénti véges közelítése

Egyetlen időpontban felvett állapot lépésenkénti véges választású közelítésének minden lépésében minden osztályt néhány, azaz véges számú újabb osztályra bontunk. Az adott időpontban felvett absztrakt állapot meghatározásakor véges számú alkalommal az osztályok alosztályjaiból választunk ki egyet. Egy lépésben az előző választás újabb lehetséges alosztályokat jelent, ezért a következő választás *típusaként* jelenik meg, míg a következő választás a típus egy *értékeként*. Az osztályokon belüli alosztályok az állapotok *típus-hierarchiáját* határozzák meg.

A választások és egyben az ismeret ábrázolásának egy lehetséges módja, ha minden választáshoz felveszünk egy megfelelő számosságú (véges) szimbólumhalmazt, a lehetőségekhez (alosztályokhoz) hozzárendelünk egy-egy szimbólumot és a választást a megfelelő szimbólummal jelöljük. Az ismeret ekkor a választott elem és a szimbólumhalmaz – mint érték és típusa – elemkettőseiből alkotott sorozattal reprezentálható. Ugyancsak az ábrázolás egy lehetséges módja, ha egy (véges) alapszimbólumhalmazból képzett véges sorozatokat rendelünk a típushierarchia egyes választási lehetőségeihez.

A választásoknak ugyancsak egy ábrázolási lehetőségét kapjuk, ha minden választáshoz hozzárendelünk egy természetes számokból alkotott szám-párt. Az elemkettős első természetes száma egy előre rögzített rendezés szerint a szimbólum sorszáma, indexét, a számpár második eleme pedig a szimbólumhalmaz számosságát jelenti. Így minden választáshoz egy változó alapszámú számrendszerben felírt törtszám egyre kisebb helyiértékű jegyeit rendeljük (pl.: $.1_{[3]} \Rightarrow .1_{[3]}3_{[4]} \Rightarrow .1_{[3]}3_{[4]}0_{[1000]}$), ahol minden alapszám a lehetséges választások száma. A kapott számjegyek a változó alapszámú számrendszerben felírt, $[0;1[$ intervallumból vett szám egyre kisebb helyiértékű jegyei ($\frac{1}{3} + \frac{3}{3 \cdot 4} + \frac{0}{3 \cdot 4 \cdot 1000}$), tehát véges számú lépéshez egy racionális szám rendelhető ($\frac{7000}{12000}$ vagy $\frac{7}{12}$). Az egymás utáni lépések a $[0;1[$ egyre szűkebb részintervallumát határozzák meg. Racionális számok-

nak ez a sorozata egy valós számhoz tart. Ha a közelítés-sorozat „határértéke” az állapot, akkor a valós szám az eredeti állapotot reprezentálja, egyébként egy, az eredeti állapotot is tartalmazó állapottípust, egy absztrakt állapotot.

Más szempontból, a kapott, változó számrendszerbeli számjegyek egy természetes szám növekvő helyiértékei, tehát egy időpontban felvett állapot adott lépésenkénti véges választású közelítése esetén egy absztrakt állapothoz egy természetes szám rendelhető (7). A számrendszer alapszámainak szorzata (mely minden lépéssel szigorúan monoton nő) pedig megadja, hogy az állapotot hány lehetőségből választottuk ki (12000).

Szimbolikus modell

A determinisztikus megadás egy kiválasztott értelmezett időpontjában felvett állapot lépésenkénti véges közelítése természetesen kiterjeszthető a teljes rendszer más időpontjainak vizsgálatára is, mellyel a rendszert tetszőleges pontossággal megközelíthetjük. A lépésenkénti véges választású közelítés feltétele alapján az állapotok típus-osztályokba történő csoportosításának egységesnek kell lennie a teljes rendszerre vonatkozóan. A pontosítás következő lépésében vagy egy új időpontban határozzuk meg adott alosztály pontosságának megfelelően az absztrakt állapotot, vagy egy már vizsgált időpont osztályát pontosítjuk annak egy alosztályára.

Általános esetben, azaz nemdeterminisztikus megadású rendszer esetén is igaz, hogy a rendszer értelmezett állapotkombinációinak számossága legfeljebb az értelmezett időpontok számossága lehet.

§ Az értelmezett állapotkombinációk számossága nem nagyobb az értelmezett időpontok számosságánál ($|\bar{S}| \leq |\bar{T}|$), illetve a két számosság pontosan akkor azonos, ha a rendszer szigorúan változó.

▽ Véges vagy korlátos rendszer. A $\sigma = (T, S, P)$ véges vagy korlátos rendszer, ha $\bar{T}$ véges.

☞ Véges rendszer esetén az $\bar{S}$ értelmezett állapotkombinációk halmaza is véges.

✓ **Véges absztrakt modell, véges modell.** A μ rendszer a σ véges vagy korlátos absztrakt modellje, ha μ véges, és egyben σ absztrakt modellje. Véges absztrakt modell példánya a kiindulási rendszer véges vagy korlátos modellje.

§ A lépésenkénti véges választású közelítés minden lépése a rendszer egy-egy véges absztrakt modelljét határozza meg.

✓ **Rendszer véges vagy korlátos szimbolikus modellje.** A $\sigma = (T, S, P)$ rendszer $\mu = (T^\Sigma, S^\Sigma, P^\mu)$ (véges) absztrakt modellje a σ véges vagy korlátos szimbolikus modellje, ha T^Σ és S^Σ véges szimbólumhalmazok, valamint T^Σ rendezett. (A σ véges szimbolikus modelljeinek jelölése: $\mathcal{M}^\Sigma \sigma$.)

☞ A σ véges szimbolikus modellje egy $\alpha_{\sqrt{T}} : T^\Sigma \rightarrow T$ és $\alpha_{\sqrt{S}} : \bar{S} \rightarrow \bar{S}^\Sigma$ függvényű típusabsztrakciója.

☞ A rendszer értelmezett időpontjainak véges részhalmazához szigorúan monoton módon, illetve állapothalmazához a természetes számok egy-egy véges részhalmazát rendelve a rendszer véges szimbolikus modelljét kapjuk a természetes számok két részhalmazát egy-egy szimbólumhalmaznak tekintve.

A véges szimbolikus modell és a következő szimbolikus modellek definíciójában a rendszerek *determinisztikus* és *egyszerűsített* megadását használjuk. Ezzel a megállapításokat az azonosnak tekintett rendszerek egész csoportjára fogalmazzuk meg.

A véges absztrakt modellek bizonyos szempontból nem zártak, mivel egy n számú időponttal és m állapotosztállyal megadott absztrakt modell pontosítása már nem feltétlenül ábrázolható egy azonos számú időponttal és állapotosztállyal megadott véges modellel. A rendszerek pontosabb megközelítését lehetővé tevő *racióális szimbolikus modell* egy adott időponthoz nem csak egy véges szimbólumhalmaz elemét (1 hosszúságú „sorozatát”), hanem a szimbólumok véges sorozatát is rendelheti, illetve magát a rendszer idejét is szimbólumok véges és rendezett sorozatával jelöli el. Véges szimbólumhalmaz tetszőleges véges hosszúságú sorozatai egy végtelen halmazt is alkothatnak, így a racionális rendszer végtelen sok időpillanaton lehet értel-

mezve és végtelen számú tényleges állapota is lehet, de minden időpont és az ott felvett állapot egy tetszőleges lépésenkénti véges választású közelítés véges számú lépése után pontosan meghatározható.

A véges és a racionális modellek között helyezkednek el a kvantált absztrakt modellek, amelyeket, hasonló elnevezéseket alkalmazva, *természetes*, illetve *integrális* (egész) absztrakt modelleknek, modelleknek, illetve szimbolikus modelleknek nevezünk. Kvantált absztrakt modell $\bar{T}$ értelmezett időpontjaihoz hozzárendelhető a szimbólumok véges sorozata, ugyanakkor egy kiegészítő feltételként azt is megköveteljük, hogy minden egyes, adott elemnél kisebb elem esetén meghatározható legyen annak a „következője”. A természetes rendszer pedig egy startpontú kvantumrendszerként határozható meg.

✓ **Természetes rendszer.** A $\sigma = (T, S, P)$ startpontú kvantumrendszert *természetes rendszernek* nevezzük.

☞ *A véges rendszer egyben természetes rendszer is.*

✓ **Természetes absztrakt modell, természetes modell.** A μ rendszer a σ *természetes absztrakt modellje*, ha μ természetes, és egyben σ absztrakt modellje. Természetes absztrakt modell példánya a kiindulási rendszer *természetes modellje*.

✓ **Integrális rendszer.** A $\sigma = (T, S, P)$ kvantumrendszert *integrális rendszernek* nevezzük.

☞ *A véges és a természetes rendszer egyben integrális rendszer is.*

✓ **Integrális absztrakt modell, integrális modell.** A μ rendszer a σ *integrális absztrakt modellje*, ha μ integrális, és egyben σ absztrakt modellje. Integrális absztrakt modell példánya a kiindulási rendszer *integrális modellje*.

§ **Kvantált és integrális absztrakt modellek.** Rendszer kvantált absztrakt modelljei pontosan a rendszer integrális absztrakt modelljei, illetve a rendszer kvantált modelljei pontosan a rendszer integrális modelljei.

✓ **Racionális rendszer.** A $\sigma = (T, S, P)$ *racionális rendszer*, ha $\bar{T}$ legfeljebb megszámlálható.

☞ *Racionális rendszer $\bar{S}$ értelmezett állapotkombinációinak halmaza is legfeljebb megszámlálható.*

☞ *A véges, természetes és integrális rendszer egyben racionális rendszer is.*

✓ **Racionális absztrakt modell, racionális modell.** A μ rendszer a σ *racionális absztrakt modellje*, ha μ racionális, és egyben σ absztrakt modellje. Racionális absztrakt modell példánya a kiindulási rendszer *racionális modellje*.

§ **Racionális absztrakt modell lépésenkénti véges választású közelítéssel történő pontosításának eredménye is racionális absztrakt modell.**

✓ **Rendszer racionális szimbolikus modellje.** A $\sigma = (T, S, P)$ rendszer $\mu = (T^{\Sigma^*}, S^{\Sigma^*}, P^\mu)$ (racionális) absztrakt modellje a σ *racionális szimbolikus modellje*, ha T^{Σ^*} és S^{Σ^*} a megfelelő T^Σ és S^Σ véges szimbólumhalmazokból képzett véges szimbólumsorozatok halmazai, valamint T^{Σ^*} rendezett. (A σ racionális szimbolikus modelljeinek jelölése: $\mathcal{M}^{\Sigma^*}\sigma$.)

☞ *A σ racionális szimbolikus modellje egy $\alpha_{\sqrt{T}} : T^{\Sigma^*} \rightarrow T$ és $\alpha_{\sqrt{S}} : \bar{S} \rightarrow S^{\Sigma^*}$ függvényű típusabsztrakciója.*

✓ **Rendszer integrális szimbolikus modellje** a rendszer olyan racionális szimbolikus modellje, amely egyben kvantumrendszer, azaz minden $t < t_g$ ($t, t_g \in T^{\Sigma^*}$) esetén létezik a t elem $t' \in T^{\Sigma^*}$ rákövetkezője.

✓ **Rendszer természetes szimbolikus modellje** a rendszer startpontú integrális szimbolikus modellje, tehát ha az értelmezett időpillanatoknak létezik minimuma.

☞ *Rendszer értelmezett időpontjainak megfelelő részhalmazához szigorúan monoton módon, illetve állapothalmazához a természetes számok részhalmazát rendelve a rendszer természetes szimbolikus modelljét kapjuk, mivel a természetes számok leírhatók egy megfelelő véges szimbólumhalmaz-*

ból képzett véges sorozatokkal, a halmazok kvantáltak, (minden adott elemnél kisebb elemnek meghatározható a következője), valamint létezik minimumértékük.

☞ *Rendszer értelmezett időpontjainak megfelelő részhalmazához szigorúan monoton módon, illetve állapothalmazához az egész számok részhalmazát rendelve a rendszer integrális szimbolikus modelljét kapjuk, mivel az egész számok leírhatók egy megfelelő véges szimbólumhalmazból képzett véges sorozatokkal, valamint a halmazok kvantáltak.*

☞ *Rendszer értelmezett időpontjainak legfeljebb megszámlálható részhalmazához szigorúan monoton módon, illetve állapothalmazához a racionális számok részhalmazát rendelve a rendszer racionális szimbolikus modelljét kapjuk, mivel a racionális számok leírhatók egy megfelelő véges szimbólumhalmazból képzett véges sorozatokkal.*

A racionális absztrakt modellek bizonyos szempontból szintén nem zártak. Bár tetszőleges lépésenkénti véges pontosítás adott lépésének eredményét ábrázolni tudják, végtelen számú pontosítás eredményét, a minden határon túlnyúló közelítések határértékét azonban nem. A rendszerek pontosabb ábrázolását lehetővé tevő *valós szimbolikus modell* az időpontokhoz és az állapotokhoz a szimbólumok tetszőleges, akár végtelen sorozatát is rendelheti.

✓ **Valós rendszer.** A $\sigma = (T, S, P)$ *valós rendszer*, ha $\bar{T}$ legfeljebb kontinuum számosságú.

☞ *Valós rendszer legfeljebb kontinuum állapotkombinációt értelmez.*

☞ *Véges, természetes, integrális, vagy racionális rendszer egyben valós rendszer is.*

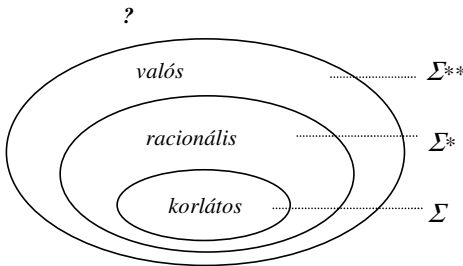
✓ **Valós absztrakt modell, valós modell.** A μ rendszer a σ *valós absztrakt modellje*, ha μ valós, és egyben σ absztrakt modellje. Valós absztrakt modell példánya a kiindulási rendszer *valós modellje*.

§ A lépésenkénti véges választású közelítés módszerét tetszőleges, akár végtelen számú lépésben alkalmazva eredményül a rendszer valós absztrakt modelljét kapjuk.

∇ Rendszer valós szimbolikus modellje. A $\sigma = (T, S, P)$ rendszer $\mu = (T^{\Sigma^{**}}, S^{\Sigma^{**}}, P^{\mu})$ (racionális) absztrakt modellje a σ valós szimbolikus modellje, ha $T^{\Sigma^{**}}$ és $S^{\Sigma^{**}}$ a megfelelő T^{Σ} és S^{Σ} véges szimbólumhalmazokból T^{Σ} és S^{Σ} véges szimbólumhalmazokból képzett tetszőleges (akár végtelen) hosszúságú szimbólumsorozatok halmazai, valamint $T^{\Sigma^{**}}$ rendezett. (A σ racionalis szimbolikus modelljeinek jelölése: $\mathcal{M}^{\Sigma^{**}} \sigma$ vagy $\mathcal{M}^{\Sigma^{\infty}} \sigma$.)

☞ A σ racionalis szimbolikus modellje egy $\alpha_{\sqrt{T}} : T^{\Sigma^{**}} \rightarrow T$ és $\alpha_{\sqrt{S}} : \bar{S} \rightarrow \bar{S}^{\Sigma^{**}}$ függvényű típusabsztrakciója.

☞ Rendszer értelmezett időpontjainak legfeljebb kontinuum számosságú részhalmazához szigorúan monoton módon, illetve állapothalmazához valós számok részhalmazát rendelve a rendszer valós szimbolikus modelljét kapjuk, mivel ezeknek a számhalmazoknak az elemei egy megfelelő véges szimbólumhalmazból képzett tetszőleges hosszúságú sorozatokkal leírhatók.



21. ábra: Korlátos, racionalis és valós absztrakt modellek.

A véges számú lehetőségéből történő lépésenkénti választás egyszerű igen/nem döntésekre vezethető vissza. A diszkrét döntések megközelítési módszere alapvetően behatárolja a vizsgált rendszerről megszerezhető információk körét és a döntések száma alapján három, minőségileg különböző szintet határoz meg. Ha egy előre megadott értékkel korlátozzuk a döntések számát, akkor a vizsgált rendszer korlátos vagy véges absztrakt modelljét kapjuk. Tetszőleges, de véges számú döntés engedélyezése a racionalis

absztrakt modellt eredményezi, míg a döntések tetszőleges, akár végtelen száma esetén a valós modellhez jutunk.

Racionalitás és analógia

A lépésenkénti véges számú lehetőségből történő választás módszerének egyenes következménye a kiindulási rendszerről alkotott racionális, illetve ennek absztrahált hataraként kapott valós kép. A racionalitás pedig közvetve a véges szimbólumhalmazból képzett véges hosszúságú sorozatokkal, a szimbolikus modellel történő megfogalmazást implikálja.

Felmerülhet a kérdés, hogy létezik-e a valóson túli megközelítési mód, mellyel az információk bővebb köre szerezhető meg, illetve amellyel felgyorsíthatjuk a vizsgált rendszer közelítését? A *választások* módszere egy kontrakciót, sőt, egy szigorúan monoton közelítést, azaz egy „ideális” ismeretszerzést ad meg. A *lépésenkénti közelítés* módosítása, például a folytonos ismeretszerzés sem ad minőségi változást, mivel a lépésenkénti esetben mindössze az a különbség, hogy az eredményeket csak bizonyos mérföldkövek esetén vizsgáljuk.

A lépésenkénti és választásokon alapuló ismeretszerzési módszerek körében maradván azonban valóban lehetséges, hogy a valós absztrakt modellt eredményező *lépésenkénti véges választások* módszerének drasztikus átalakításával bővítsük a megszerezhető információk körét, illetve felgyorsítsuk a vizsgált rendszer közelítését. Erre pedig mindössze egyetlen lehetőségünk van: ha lépésenként végtelen, akár kontinuum számosságú lehetőségből történő választást is lehetővé teszünk!

Korlátai ellenére a racionális megközelítés több előnnyel is rendelkezik. Lassúsága ellenére a rendszer *tetszőleges pontosságú megközelítésére* is alkalmas. Bár ezt a fogalmat mindössze abban az értelemben használtuk, hogy minden, még nem egzakt közelítésnél a következő lépések pontosabb ismeretet eredményeznek. Másik előnye, hogy a közelítések könnyen dokumentálhatók, mivel lépésenként mindössze néhány, azaz véges számú lehetőségből választunk, így a választások sorozata mások által is nyomonkövethető és ellenőrizhető.

A logika és a racionalitás szavak azonos fogalomkörre utalnak. A „racionális” egy természetesen megfogalmazható és ellenőrizhető arányosítás, azaz egy speciális viszony jelzője, amely a matematikában két egész szám aránya-

ként (vagy két természetes szám előjelezett arányaként) jelenik meg. Az ókori görögök a természetes eszközökkel meghatározható arányt „logos”-nak (λογος) nevezték, amelyből maga a logika, illetve a „logikus” szó is ered, és amely a későbbiekben a „racionális” szóval együtt az ellenőrizhető ismeretek jelzésére szolgált. „Természetes eszközön”, például a „természetes módú ellenőrzésen” olyan eljárást értünk, amely adott kezdőpontból kiindulva véges (azaz egy *természetes számmal* megadható) számú, eleminek tekinthető lépésekben eredményt ad. Ezért a „természetes” jelző is a racionális, illetve a logikus jelzőkkel azonos fogalomkörre utal.

Amikor az ókori görögök egy olyan viszonyt találtak, amely természetes eszközökkel, azaz „logos”-ként nem volt megfogalmazható, azt „analogos”-nak nevezték, mely az *analógia* szavunk eredete. A „nem logikai elvű”, azaz „analógiás” jelző így jelölheti a végtelen számú lehetőségből történő választáson alapuló ismeretszerzést, illetve ismeretreprezentációt. Az analógnak nevezett műszerek, áramkörök, stb. maguk is végtelen számú lehetőségből történő választást valósítanak meg.

Hogyan képzelhető el egy analógiás, azaz végtelen számú lehetőségből történő választáson alapuló ismeretszerzés? Tekintsük a megismerendő rendszer minden egyes állapotát egy-egy összetett hullámként. A rezonancia, amely a hullámforma fő vonalait ábrázolni képes, végtelen számú lehetőségből választ. A hullámforma fő vonalain érthetjük az összetett forma Fourier-sorfejtésének első (vagy első néhány) tagjának meghatározását, amely végtelen számú lehetőségből történő választást, a megfelelő valós számok megadását jelenti. Az ismeretszerzés következő lépései további tagokat határozhatnak meg. Egy forma Fourier-sorfejtésének kezdő tagjai tekinthetők a forma típusabsztrakciójának.

Az analógiás ismeretszerzés tehát elképzelhető egy „ráhangolódásként”, egy rezonancia-elven működő monoton közelítésként. Az ismeret ábrázolása megismertet az ismerettel, így az analógiás ábrázolásnak reprezentálnia kell a végtelen lehetőségek közötti választást. Ez a reprezentáció elképzelhető úgy, mint ami az interpretáció során maga is összetett hullámformát gerjeszt. Például egy vers hangzókaváltsái, szavaihoz, szóösszetételeihez kapcsolódó formák a kiolvasás során egy összetett „hullámformát” generálhatnak, melyek a megfelelő sorrendbe helyezve analógiás módon ábrázolni képesek az ugyancsak „hullámformaként” tekinthető rendszert.

Az analógiás megközelítés egy gyorsabb és teljesebb körű ismeretszerzést tesz lehetővé, de az így szerzett bonyolult ismeretek teljeskörűen csak analógiás eszközökkel ellenőrizhetők, racionális eszközökkel nem. Az analógiás megközelítés ezért a racionalitás számára a bizonytalanságot, az ellenőrizhetetlenséget jelenti. Például még az sem feltétlenül megfogalmazható, hogy egy bonyolult jelölési rendszer *ténylegesen* milyen „hullámokat” gerjeszt.

Ha egy ismeretszerzési technika csak kevés számú lehetőségből (például néhány ezerből) történő választást tesz lehetővé, akkor a nagyon *nagyszámú* (például néhány millió vagy milliárd) lehetőség a gyakorlatban végtelenként jelenik meg. A gyakorlatban a választóvonal nem az elméleti véges és végtelen között, hanem a még áttekinthető *néhány* és a gyakorlatilag áttekinthetetlenül *nagyszámú* lehetőség között húzódik.

Az ábrázolás és így az ismeretszerzés lényegileg a lehetőségek közül történő választás technikáján alapszik, ezért ezek két, minőségileg különböző csoportba sorolhatók. A gyakorlatilag végtelen számú lehetőségből történő választás engedélyezése az analógia világát jelenti, míg a véges, áttekinthető számúvá korlátozott lehetőségek az ellenőrizhetőséget, a racionalitást. A véges számú lehetőségből történő választás, azaz az ellenőrizhető döntések világa további három alcsoportra osztható fel, mely alcsoportok a véges, a racionális, illetve ennek határértékeként kapott valós ábrázolás.



Hatás és kapcsolat

Az absztrakt megadási mód egyetlen definícióval a működések egész csoportjának a definiálására alkalmas. A „változó változás” vizsgálatával megállapítható, hogy a belső összefüggések a rendszer mely részeit, milyen szempontból és milyen mértékben korlátozzák, vagy másképpen fogalmazva: a rendszer egyes részeinek változása hogyan hat más részek működésére. A rendszer egy entitásának szempontjából megállapíthatjuk annak bemente és kimenete által alkotott környezetét, valamint az ahhoz nem kapcsolódó külvilágát.

Absztrakt megadási rendszer

Eddigi eszközeink, például a hasonlítható absztrakció és pontosítás segítségével történő felbontás és ábrázolás technikái alapvetően egyetlen konkrét vagy absztrakt módon lezajló változásra alkalmazhatók. Ahhoz, hogy egy részrendszer „jellegét”, különböző környezetekbe helyezve viselkedésének jellemző jegyeit megragadhassuk, ahhoz magát a változás függvényét kell absztrakt módon megfogalmaznunk. A változás absztrakt megadásának vizsgálatával felderíthetők a rendszer részei közötti kapcsolatok, egymásra-hatások: a belső rendszer.

Eddigi rendszereink egyetlen, rögzített módon lezajló változást adtak meg. Ha egyetlen meghatározással rendszerek egész csoportjának a változását le szeretnénk írni, akkor magát a változás megadását kell általánosítanunk. Általános esetben egy absztrakt megadási rendszer egy $f: \mathcal{A} \rightarrow \{\sigma_{i \in I}\}$ függvény, amely egy $a \in \mathcal{A}$ argumentumhoz a megfelelő $f(a) = \sigma_i$ rendszert rendeli.

Az absztrakt megadási mód alkalmas a „változó változás”, azaz egy argumentumtól függően eltérő módozatú változás megadására.

Ha az absztrakt megadási módot *hasonlítható rendszerek* adott csoportjának megadására korlátozzuk, akkor az absztrakt megadású rendszer felírható egy $f : \mathcal{A} \rightarrow \Sigma^{(\sigma)^*}$ függvényként. Hasonlítható $\Sigma^{(\sigma_{i \in I})^*}$ rendszerek esetén rögzíthető egy keretül szolgáló $\mathcal{T}_\Sigma \supseteq T_\Sigma = \bigcup_{i \in I} T_i$ lehetséges idő és $S_\Sigma \supseteq \bigcup_{i \in I} \overline{S_i}$ állapothalmaz, amelyek minden rendszer számára közősek, így a hasonlítható absztrakt megadású rendszer a $(\mathcal{T}_\Sigma, S_\Sigma, P^{(\mathcal{A})})$ formában is felírható, ahol $P^{(\mathcal{A})} : \mathcal{A} \rightarrow \{P_i : T_\Sigma \rightarrow 2^{S_\Sigma} \setminus \{\emptyset\}\}$ minden argumentumhoz meghatározza a megfelelő σ_i változásának a függvényét.

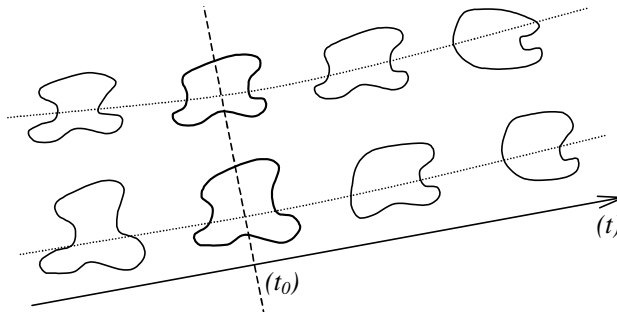
A következőkben absztrakt megadású rendszeren ezt a speciális változatot, a *közvetlenül hasonlítható rendszerek* adott csoportjának változását leíró formát értjük. A „hasonlítható absztrakt megadású rendszerre” vonatkozó eredmények egy „általános absztrakt megadású rendszerre” is kiterjeszthetők. Ehhez mindössze egy kiegészítő függvény felvétele szükséges, amely minden egyes argumentumhoz megad egy-egy átjelölést ($g : \mathcal{A} \rightarrow \{\alpha_{\equiv i \in I}\}$).

✓ Absztrakt megadású rendszer, működés. Közvetlenül hasonlítható rendszerek $\sigma^{(\mathcal{A})} = (\mathcal{T}, S, P^{(\mathcal{A})})$ *absztrakt megadása* esetén $P^{(\mathcal{A})}$ minden $a \in \mathcal{A}$ argumentumhoz egy-egy $P^{(\mathcal{A})}(a) : T \rightarrow 2^S \setminus \{\emptyset\}$ (rövidítve: $P^{(a)}$) változást rendel, melyet az absztrakt megadású rendszer *lefutásának* vagy *működésének* nevezünk.

Egy absztrakt megadású rendszer argumentumai tetszőleges halmazt alkothatnak, lehetnek például az egész, a valós, a komplex számok, vagy bármely más, nem számjellegű halmaz is. Általános esetben az argumentumok tehát a rendszer világán „kívül is állhatnak”. Éppen ezért különlegeseek azok az absztrakt megadású rendszerek, amelyek argumentumai a rendszerek bizonyos állapotkombinációi; a hasonlítható rendszereket például egy kijelölt t_0 időponthoz viszonyított s_0 állapotkombinációval is meghatározhatjuk. Ebben az esetben a $P^{(S_0)}$ formájú függvény minden t időponthoz megadja a t_0 -ban felvett s_0 állapotkombinációjú rendszer $s_t^{(s_0)} = P^{(S_0)}(s_0)(t)$ aktuális állapotkombinációját, ahol S_0 az absztrakt megadású rendszer t_0 -ban felvehető állapotkombinációinak halmazát jelöli. Ennek a megadásnak

tehát az a különlegessége, hogy az argumentumok is a rendszer világából származnak, annak egy-egy szituációi, helyzetei.

▽ **Átmeneti függvény.** Egy kitüntetett t_0 időpillanatban felvett állapotkombinációval (determinisztikus esetben: állapottal) parametrizált $\sigma^{(S_0)}$ ($S_0 \subseteq \bar{S}$ az értelmezési tartomány) absztrakt megadási $P^{(S_0)}$ rendszert *átmeneti függvénynek* nevezzük. Az S_0 az absztrakt megadási rendszer t_0 -ban felvehető állapotkombinációinak a halmaza.



22. ábra. Az átmeneti függvény az s_0 -tól függően több rendszer változását is megadja

▽ **Tiszta (reguláris) átmeneti függvény.** A $P^{(\rightarrow)}: \bar{S} \rightarrow \bar{S}$ módon megadott és a t_0 -hoz tartozó $s_{t_0} \subseteq \bar{S}$ állapotkombinációval parametrizált startpontú kvantumrendszert tiszta (más néven: reguláris) átmeneti függvénynek nevezzük, ahol a $P^{(\rightarrow)}: \bar{S} \rightarrow \bar{S}$ minden s állapotkombinációból meghatározza a következő időpont s' kombinációját, amennyiben az időpont létezik. Az s kiindulási állapotkombinációból a rendszer sorra az $s' = P^{(\rightarrow)}(s)$, $s'' = P^{(\rightarrow)}(s') = P^{(\rightarrow)}(P^{(\rightarrow)}(s))$, ... kombinációkba vált át.

§ **A tiszta átmeneti függvény egy átmeneti függvény,** mivel a tiszta átmeneti függvénnyel megadott startpontú kvantumrendszer átírható a t_0 időpillanatban felvett állapotkombinációval parametrizált átmeneti függvény formába (összetettfüggvény-képzések segítségével).

Példa: Fibonacci-számok absztrakt megadása.

Az egymás után következő t időpillanatokban az adott és az előző Fibonacci-számokat felvevő rendszer egy $s=(a,e)$ állapotból az $s'=(a+e,a)$ összefüggéssel leírható állapotba lép.

Rendszerek absztrakt megadása esetén az a célunk, hogy *több* hasonlítható rendszer változását *egyetlen* definícióval határozzuk meg. A hasonlítható rendszerek csoportja (halmaz) a rendszerek hasonlítható absztrakciójaként is tekinthető. A diszjunkció egyszerű csoportosításának eredménye azonban csak egy absztraktum, amely elveszti a konkrét változásokra vonatkozó információkat, és amelyek visszanyeréséhez egy pontosítást kell végrehajtanunk. A pontosításhoz felhasználhatjuk például az *egyszerű pontosítás* technikáját. Ekkor az absztraktumhoz egy paraméterszerepű új, konstans *állapotkomponenst* adunk, amellyel egyenként meghatározhatjuk az absztrakcióban benne foglalt minden egyes változatot. Az absztrakt rendszer ezeket a pontosító paramétereket saját állapotainak bővítéseként érzékeli. Az eredményül kapott rendszer paraméterként szolgáló állapotkomponense ugyancsak rögzíthető a kitüntetett t_0 állapotban, így a paraméterekkel megfogalmazott P szintén megadható az átmeneti függvények formájában.

§ Absztrakt megadású rendszer megadható átmeneti függvénnyel.

Minden hasonlítható pontosítás megadható egyszerű pontosításként, ezért az absztrakt módon megadott és hasonlítható absztraktumként értelmezett változás pontosítása is megadható paraméterjellegű kiterjesztésként. A paraméter a rendszer minden időpontjában változatlan, konstans állapotú rész, ezért a t_0 -ban felvett s_0 állapotkombináció – benne a paraméterszerepű komponenssel – felhasználható $P^{(S_0)}$ konkretizálására.

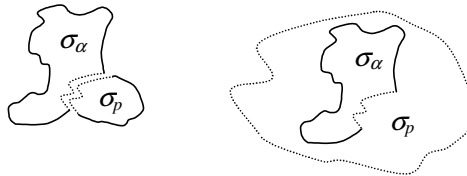
§ Átmeneti függvény módszerként. A t_0 -ban felvett s_0 állapotkombinációk egy-egy helyzetet, szituációt határoznak meg, amelyek a $\sigma^{(s_0 \in S_0)}$

rendszerekkel hasonlíthatók. Így egy átmeneti függvénnyel megadott absztrakt megadású rendszer megadható egy egyoperandusú módszerként.

Egy átmeneti függvény formájában megadott rendszert vizsgáljunk meg az argumentumként szolgáló, t_0 -ban felvett különböző állapotkombinációk alapján! Ha a rendszer egy kompozit, akkor annak komponensei gyakran két csoportra oszthatók. Egyik csoportba sorolhatjuk a minden s_0 -ban azonos

részállapotkombinációt felvevő, azaz t_0 -ban bármely paraméter esetén azonos állapotkombinációjú komponenseket, amely komponenseket együttesen most *absztrakt* komponensnek nevezzük. A másik csoport komponensei a kijelölt időpontban más állapotkombinációkat is felvehetnek, mely komponenseket együttesen egy (t_0 -ban változó) *konkretizáló* komponensként tekintünk. A két komponensre bontott rendszerben a mindig azonos kiindulási állapotkombinációjú absztrakt komponensnek a kijelölt időponttól eltérő változását a másik komponens állapotkombinációja konkretizálja.

A konkretizáló komponens, mint egy, a folyamat során végig „érzékelhető” elem kiemelhető a függvény megadásába. A kapott $P^{(S',p')}(t)$ alakú függvény újabb egyszerű átalakításával maga az s'' rész-kezdőállapot is meghatározható a paraméterek esetleges kiegészítésével, így egy $P^{(p)}(t)$ (p paraméterek) *parametrizált formát* kapunk. Parametrizált forma esetén egy, a különböző lefutások esetén a t_0 időpontban azonos és egy paraméterjellegű változó komponenst kapunk. A rendszernek ez a felbontása szemléletesen úgy is tekinthető, mintha az alapkompone ns részrendszerét különböző, paraméterekkel megadott „környezetekbe” helyeztük volna. A parametrizált megadású rendszerek tekinthetők automataként, amelyek az absztrakt rész környezetét, paraméterjellegű komponensét feldolgozva eredményül egy-egy adott lefutású rendszert eredményeznek.



23. ábra: Absztrakt megadású rendszer paramétere mint környezet.

Lényeges, hogy különbséget tegyünk absztrakt rendszer és absztrakt *megadású* rendszer között. Mindkettő absztrakcióként több rendszert foglal magába. De míg általánosítás esetén elvesztjük a folyamatok pontos lefolyására vonatkozó információt, addig az absztrakt megadás az argumentum meghatározásával konkretizálható.

Példa: NDFA: feltételes utasítás és párhuzamosság.

Az absztrakt és az absztrakt megadású rendszerekre példa egy nemdeterminisztikus automata, amely változása során „vagy” jellegű absztrakt

makroállapotba mehet át: a vezérlés „vagy” az egyik, „vagy” a másik ágon folytatódik. A pontosító paraméterek a folyamat konkrét lezajlásának ágát választhatják ki. A programozási nyelvekben ez a feltételes végrehajtásnak felel meg, ahol egy paramétertől függően egyik vagy másik kódrészlet hajtódik végre, így konkretizálódik az eredetileg absztrakt változás.

§ Minden rendszer tekinthető absztrakt megadású rendszerként. Ebben az esetben minden $a \in \mathcal{A}$ argumentumértékhez azonos működés tartozik, például, ha az $\mathcal{A}$ halmaz egyetlen elemű.

Az absztrakt megadási módokkal adott szabályokat teljesítő rendszerek egész csoportját tudjuk átfogni. Véges számú rendszer esetén megtehetnénk, hogy egyszerűen felsoroljuk a lehetséges változatokat, ekkor azonban gyakran részben vagy teljesen ugyanazok az állapotkombinációk fordulnának elő. Ez a redundancia megnehezíti mind a rendszerek megadását, mind a megadások megértését és módosítását. Az absztrakt szemlélet a szabályok felismerésével egyetlen, paraméterekkel konkretizálható absztraktumként képes megfogalmazni a rendszerek csoportját. Így a redundancia csökkenthető, optimális esetben el is tüntethető és a működést egy rövidebb és kezelhetőbb módon adhatjuk meg.

Átírás nem absztrakt megadású rendszerre

A nem absztrakt megadású rendszerekre vonatkozó fogalmakat absztrakt megadású rendszerekre is értelmeznünk kell. Ehhez megadunk egy olyan megfeleltetést, amely minden absztrakt megadású rendszerhez egyértelmű módon egy nem absztrakt megadású rendszert rendel. Így az absztrakt megadású rendszereket nem absztrakt megadású rendszerre átírva az eredeti fogalmak teljes apparátusát használhatjuk. Az átalakítás során az egyes $a \in \mathcal{A}$ argumentumértékekhez tartozó $P^{(a)}$ működéseket úgy tekintjük, mintha az egy kompozit más és más részeinek változását határozná meg. Elvégezhető ez az átalakítás, mivel a (hasonlítható) absztrakt megadású rendszerek időpontjainak halmaza azonos.

▽ Absztrakt megadású rendszer értelmezése nem absztrakt megadású rendszerként. A $\sigma^{(\mathcal{A})} = (\mathcal{T}, \mathcal{S}, P^{(\mathcal{A})})$ absztrakt megadású rendszert

nem absztrakt megadási mód esetén a $\sigma = (T, \otimes_{a \in \mathcal{A}} S, P)$ rendszerként tekintjük, ahol $P_{(a)} = P^{(a)}$, azaz az a rész működését az absztrakt megadási mód a argumentumú rendszere írja le.

☞ *Absztrakt megadású rendszer nem absztrakt megadású rendszerré való átírása egyértelmű.*

Az átírás segítségével az absztrakt megadású rendszerek közvetlen hasonlíthatósága olyan rendszereket jelent, melyek idő és állapothalmaza mellett az $\mathcal{A}$ argumentumhalmazok is azonosak. A (közvetve) *hasonlítható rendszerek* esetén az argumentumok halmaza bővíthető, vagy abból elhagyhatók olyan értékek, melyekhez minden időpontban bizonytalan állapotú, azaz tetszőleges absztrakt működés tartozik.

Az absztrakt megadású rendszerek hasonlítható működéseket határoznak meg, így a *kompozícióval* kapcsolatos fogalmak absztrakt megadású rendszerek esetén közvetlenül használhatók. Például egy absztrakt megadású rendszer vizsgálható csak bizonyos komponensek esetén, illetve elvonatkoztathatunk adott komponenseitől.

Absztrakt megadású rendszer *hasonlítható absztrakciója* egy olyan absztrakt megadású rendszert eredményez, melyben minden egyes működés a megfelelő eredeti működés hasonlítható absztrakciója. A hasonlítható absztrakció során elhagyhatunk, tetszőleges absztrakt működésűvé változtathatunk adott működéseket, illetve felvehetünk olyan új argumentumértékeket, melyekhez tetszőleges absztrakt működés tartozik. Absztrakt megadású rendszer *korlátozója*, így legalább egy működést korlátoz, illetve más működéseket különböző módon is korlátozhat.

Absztrakt megadású rendszer korlátozóinál erősebbeknek tekinthetők azok a korlátozók, amelyek néhány vagy az összes működést (amelyek közvetlenül hasonlíthatók) egyszerre korlátoznak. A *működés korlátozója* az értelmezett időpontjaiban minden vizsgált működés felvehető állapotkombinációit szűkíti, azaz az összes vizsgált működésre vonatkozó információ.

Működések feltétele és korlátozója

Absztrakt megadású rendszer esetén a működésekkel hasonlítható valamely rendszer (ami a működésekre vonatkozó ismeret) kiválogatja azon működések, amelyekre az ismeret következményként (hasonlítható absztrakcióként) teljesül. A működések részalmazának ez a kiválasztása egyben az $\mathcal{A}$ argumentumhalmaz egy részalmazának a kiválasztását is jelenti. Az argumentumok tetszőleges részalmazát a következőkben feltételargumentumoknak vagy röviden feltételnek nevezzük, mivel az kiválasztja az „adott feltételeknek megfelelő” argumentumok részalmazát.

Ha az ismeret a tetszőleges absztrakt rendszer, akkor az ahhoz tartozó feltételargumentum-halmaz megegyezik a teljes argumentumhalmazzal, mivel egy tetszőleges absztrakt rendszer bármely rendszer következménye. Ha az ismeret egyetlen működésnek sem a következménye, például az összes működésnek ellentmond, akkor az ehhez a következményhez tartozó feltételargumentum-halmaz üres.

✓ **Absztrakt megadású rendszer feltételargumentumai.** A $\sigma^{(\mathcal{A})}$ absztrakt megadású rendszer argumentumhalmazának $\mathcal{C} \subseteq \mathcal{A}$ részalmazát a $\sigma^{(c)}$ ($c \in \mathcal{C}$) működések feltételének vagy feltételargumentumainak nevezzük, a $\sigma^{(c)}$ ($c \in \mathcal{C}$) lefutásokat pedig a $\mathcal{C}$ feltételű működéseknek.

§ **Adott feltételű működések mint absztrakt megadású rendszer.** Absztrakt megadású rendszer adott feltételű működései együttesen ugyan csak egy absztrakt megadású rendszert alkotnak, amely az eredeti absztrakt megadású rendszer leszűkítése, így az adott feltételű működések vizsgálhatók feltétel nélküli absztrakt megadású rendszerként, illetve használható a $\sigma^{(\mathcal{C} \subseteq \mathcal{A})}$, illetve rövidítve a $\sigma^{(\mathcal{C})}$ jelölés.

✓ **Adott feltételű és feltétlen absztrakt működés.** Absztrakt megadású rendszer adott $\sigma^{(\mathcal{C} \subseteq \mathcal{A})}$ működéseinek $\bigcup_{c \in \mathcal{C}} \sigma^{(c)}$ unióját a $\mathcal{C}$ *feltételű absztrakt működésnek* (jelölés: $\bigcup \sigma^{(\mathcal{C})}$) vagy *feltételes absztrakt működésnek* nevezzük. Ha a feltétel a teljes argumentumhalmaz, akkor az absztrakt megadású rendszer összes lefutásának $\bigcup_{a \in \mathcal{A}} \sigma^{(a)}$ unióját (jelölés: $\bigcup \sigma^{(\mathcal{A})}$) *feltétlen absztrakt működésnek* nevezzük.

∇ Működések feltételes és feltétlen korlátozója és következménye.

Absztrakt megadású rendszer $\sigma^{(C \subseteq A)}$ működéseinek korlátozója a σ' , ha az minden egyes $\sigma^{(c)}$ ($c \in C$) működés korlátozója, illetve a *működések következménye* a σ'' (jelölés: $\sigma'' \Leftarrow \sigma^{(C \subseteq A)}$), ha $\forall c \in C: \sigma'' \Leftarrow \sigma^{(c)}$. A működés korlátozója, illetve következménye *feltétlen*, ha $C = A$, egyébként (valódi részhalmaz esetén) a működések korlátozója, illetve a működések következménye *feltételes*.

☞ *Adott működések következménye az adott működések korlátozója vagy a tetszőleges absztrakt rendszer.*

Megkülönböztetjük tehát az absztrakt megadású rendszer feltételargumentumait, vagy röviden *feltételét*, illetve az *adott feltételű absztrakt működést* (amely a feltételargumentumok által meghatározott működések uniója), valamint adott működések *korlátozó feltételét* (a vizsgált működések olyan következményét, amely nem tetszőleges absztrakt rendszer). A két utóbbi fogalom szoros kapcsolatban van egymással.

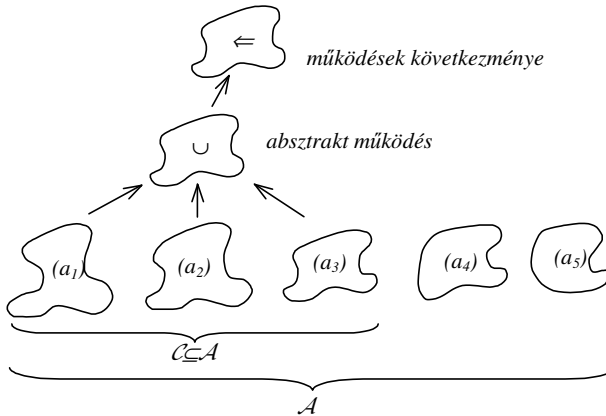
§ **Adott feltételű működéseknek a σ' pontosan akkor a következménye, ha az az adott feltételű absztrakt működés következménye, azaz**
 $(\sigma' \Leftarrow \sigma^{(C \subseteq A)}) \Leftrightarrow (\sigma' \Leftarrow \bigcup \sigma^{(C \subseteq A)}).$

Az absztrakt megadású rendszer működéseivel hasonlítható rendszer, azaz a működésekre vonatkozó *ismeret* egyben kiválasztja azon működéseket, amelyeknek az ismeret a következménye. Így az ismeret az argumentumok egy részhalmazát, egy feltételargumentum-halmazt határoz meg. Azokat az argumentumokat, amelyek esetén az ismeret igaz (azaz következmény), az ismeret feltételének nevezzük.

∇ Működésekre vonatkozó ismeret és következmény feltétele.

A *működésekre vonatkozó σ' ismeret feltétele* az argumentumok $C_{\sigma'} = \{c \in A \mid \sigma' \Leftarrow \sigma^{(c)}\}$ részhalmazára (ami a $\sigma^{(A)}$ feltétele).

Egy vagy több működés σ' következménye bizonyos $\sigma^{(C_{\sigma'})}$ működéseket határoz meg, tehát egy következmény alkalmas több működés együttes megadására.



24. ábra. Működések tartománya, ill. következmény feltételargumentumai

☞ A működésekre vonatkozó σ' ismeret feltétele üreshalmaz pontosan akkor, ha a σ' egyetlen működésnek sem következménye.

☞ A működésekre vonatkozó σ' ismeret feltétele a teljes $\mathcal{A}$ argumentumhalmaz pontosan akkor, ha a σ' a működések feltétlen következménye.

☞ A működésekkel hasonlítható tetszőleges absztrakt rendszer a működések feltétlen következménye.

Entitás korlátozója

A rendszer, azaz a változást meghatározó P függvény különböző *korlátozó megszorításokként* jelenik meg, információként, amely szűkíti a lehetőségeket, azaz a lehetséges értékekből választ ki egy-egy részhalmazt. Kiválasztásként tekinthető például az, ahogy a rendszer összes $2^S \setminus \{\emptyset\}$ felvehető állapotkombinációiból a P kiválasztja annak egy részhalmazát ($\bar{S} = P(T)$). A lehetséges állapotkombinációkat az értékkészlet leszűkíti a *ténylegesen felvett állapotkombinációk* halmazára. A változás függvénye mivel meghatározza, ezért korlátozza az értelmezett időpontokban a felvehető állapotkombinációkat.

▽ Relációs vagy tartomány-, és funkcionális- vagy értékkorlátozó.

Rendszer korlátozója *funkcionális-* vagy *értékkorlátozó*, ha az példány. A nem értékkorlátozót *relációs*, *értéktartomány-* vagy *tartománykorlátozó*nak nevezzük.

Egy értékkorlátozó tehát egyetlen elemre, egyetlen értékre szűkíti a felvehető állapotokat, míg a tartománykorlátozó több felvehető értéket is megenged.

A következőkben entitásnak nevezzük egy kompozit valamely meghatározott részét, a definíciókat és állításokat pedig ilyen entításokra fogalmazzuk meg.

▽ Entitás. A $\sigma = (T, L, \overline{T \times L}, S, P)$ rendszernek a $\pi_E \in 2^L \setminus \{\emptyset\}$ entitás-azonosítójú E entitása a σ leszűkítése az $l \in \pi_E$ pozíciókra. A $T_E = \{t \mid (t, l) \in \overline{T \times L} \wedge l \in \pi_E\}$ az *entitás élettartama*. Az entitás a $t \in T_E$ időpontban $E(t) = \otimes_{l \in \pi_E} P(t, l)$ állapotban van, a többi $t \in T \setminus T_E$ időpontban pedig bizonytalan állapotú.

Mivel egy kompozit egyben önmaga része is, ezért az entításra vonatkozó fogalmak és állítások automatikusan a teljes rendszerre is alkalmazhatók.

Absztrakt megadású rendszer egy entitását vizsgálva az különböző argumentumok esetén különböző módon is változhat.

Egy entitás bizonyos működéseinek korlátozása azt jelenti, hogy adott időpontokban az entitás által elfoglalt π_E helyeken a felvett állapot-kombinációkat szűkíteni tudjuk, azaz a tetszőlegességet jelentő $S_{|\pi_E}$ halmaznak egy valódi részhalmazát határozhatjuk meg. Az állapotlehetőségekből a valódi részhalmazok kiválasztása korlátozó, mely az entitás és egyben a rendszer hasonlítható absztrakciója lesz, ugyanakkor nem tetszőleges absztrakt rendszer. Ez a korlátozó az entitás és egyben a teljes rendszer adott feltételű működéseire vonatkozó információ lesz.

▮ **Entitás korlátozója.** Kiválasztott időpontokra és egy entitásra leszűkített absztrakt megadású rendszer adott működéseinek korlátozóját az *entitás (működésének) adott időpontokra vonatkozó korlátozójának* nevezzük.

☞ *Entitás korlátozója egyben a teljes rendszer adott működéseinek korlátozója, ami feltétlen korlátozó esetén minden egyes működésre vonatkozó információ.*

Mivel a teljes rendszer is tekinthető entitásként, ezért az entitás korlátozója elnevezés használható a működések korlátozója szinonimjaként is. Az entitáskorlátozó elnevezés azonban arra is utal, hogy az absztrakt megadású rendszert egy entitására szűkítve vizsgáljuk.

Absztrakt megadású rendszernek a változás $P^{(\mathcal{A})}$ függvénye egy entitást adott időpontokban vagy korlátozatlanul hagy, így az különböző $a \in \mathcal{A}$ értékek esetén bármely állapotban lehet, vagy a felvehető állapotait egy tartomány- vagy értékkorlátozóval egy valódi részhalmazra korlátozza.

▮ **Entitás működéseinek tartománya.** A $\sigma^{(C)}$ működések E entitása a vizsgált $T' \subseteq T$ időpontokban $E_{T'}^{(C)} = \sigma_{|ET'}^{(C)}$ módon változik. Az E entitás $E_{T'}^{(C)}$ működéseinek tartománya az E entitás $\bigcup E_{T'}^{(C)}$ absztrakt működése.

Bővebb argumentumhalmaz esetén az entitás változatosabb is lehet, ezért az absztrakt működése és így egyben a működéseinek a tartománya is bővíülhet:

§ **Bővebb feltétel esetén a tartomány bővül.** $C' \subseteq C$ esetén $\bigcup E_{T'}^{(C')} \subseteq \bigcup E_{T'}^{(C)}$.

Absztrakt megadású rendszer esetén adott működéseket kiválasztva a változás függvénye az entitás szempontjából különböző korlátozóként jelenhet meg, mivel egy kiválasztott időpontban vagy időpontokban a tetszőlegességnél szűkebb állapotlehetőségeket határozhat meg.

§ **A változás függvénye mint korlátozó.** Az $E^{(C)}$ működésű entitást a $T' \subseteq T$ időpontokban vizsgálva a változás függvénye az időpontokban

korlátozza az entitás működését (jelölés: $\rightarrow_T^{(C)} E$), ha az $\bigcup E_{T'}^{(C)}$ nem tetszőleges absztrakt rendszer (hanem az az E korlátozója), ellenkező esetben az entitás az időpontokban korlátozatlan (jelölés: $\nrightarrow_T^{(C)} E$). Az entitás korlátozója lehet érték- vagy tartománykorlátozó is.

✓ **Strukturális és időleges tulajdonság.** Ha egy tulajdonság a rendszer összes értelmezett időpontjában fennáll, akkor azt *strukturálisnak* nevezzük, míg ellenkező esetben, ha találhatók olyan értelmezett időpillanatok, amikor a tulajdonság fennáll és olyanok is, amikor a tulajdonság sérül, akkor a tulajdonságot *időlegesnek* (temporálisnak) nevezzük. Strukturális tulajdonság jelölésekor elhagyható az összes értelmezett időpontok halmazának jelölése (pl. entitás strukturális korlátozója esetén: $\rightarrow^{(C)} E$).

Entitáskorlátozók erőssége

Egy vizsgált entitás működéseinek következményei között meghatározható egy „erősség” viszony. Két függő következmény esetén az „erősebb” a felvehető állapotoknak szűkebb részhalmazát határozza meg, míg a „gyengébbnek” nevezhető több állapotlehetőséget enged meg. A „legerősebb” korlátozók a példányjellegű értékkorlátozók, melyek tovább már nem korlátozhatók. A „leggyengébb következményeknek” pedig a korlátozatlan állapotok, az entitás összes állapotlehetősége tekinthető, amely már tetszőleges absztrakt rendszer.

✓ **Működés következményének gyengesége.** Absztrakt megadású rendszer $\sigma^{(C \subseteq A)}$ működései közvetlenül hasonlíthatók, ezért rögzíthető egy $\prec$ szempont. Vegyük a működéseknek egy $\sigma' \leftarrow \sigma^{(C)}$ következményét (ha a σ' nem a hasonlítható tetszőleges absztrakt rendszer, akkor a működések korlátozója). A $\sigma^{(C)}$ működések σ' következményének $\prec$ szempont szerinti *gyengeségének* azt a legnagyobb x értéket értjük, melyre bármely $\sigma^{(c)}$ ($c \in C$) működés $\prec_{(x)} (\sigma^{(c)})$ absztrakciójának a σ' még a következménye.

A szempont a működések következményeit állítja elő, mely következmények egy adott érték fölött a vizsgált következményt „túlhaladhatják”, azaz absztraktabbá válhatnak, vagy attól „eltérhetnek”. Mindkét esetben a

működések szempont szerinti hasonlítható absztrakciói bizonyos időpontokban olyan állapotokat is megengednek, melyeket a következmény nem tartalmazott.

☞ *A hasonlítható absztrakt működés maximális gyengeségű, az minden más következménynél gyengébb.*

§ Rögzített működések következményeinek gyengesége egy következmény-rendezést határoz meg. Ha $\sigma'' \Leftarrow \sigma' \Leftarrow \sigma^{(C \subseteq A)}$, akkor bármely szempont esetén a σ'' gyengesége nagyobb a σ' gyengeségénél. Így a működések függő következményei esetén beszélhetünk a *működés gyengébb és erősebb következményeiről*.

§ Működések következményének hasonlítható absztrakciója a működés gyengébb következménye.

☞ *Működés korlátozó erősebbek a hasonlítható absztrakt működésnél.*

§ Az absztrakció gyengíti a korlátozókat, azaz adott működések következményénél a működések hasonlítható absztrakcióira gyengébb következmény adható meg. Adott $\bar{\sigma}$ következményű $\sigma^{(C)}$ működések az $\sigma'^{(C)}$ hasonlítható absztrakcióira cserélve lehetséges, hogy a $\bar{\sigma}$ már nem minden $\sigma'^{(C)}$ működés következménye, de található a $\bar{\sigma}$ olyan hasonlítható absztrakciója, amely már a $\sigma'^{(C)}$ működések következménye (például ilyen a hasonlítható absztrakt rendszer is). Működések hasonlítható absztrakciója ezért gyengítheti a működések következményeit.

Hatás, függés, kapcsolat

A rendszer egyes részei, entitásai közötti hatás úgy fogalmazható meg, hogy a ható entitás változása kikényszeríti a függő entitás változását, így a ható entitás korlátozza, „adott mederbe tereli” a függő entitás változását. Egy A entitás hatása egy B entításra a két entitás működése közötti „*ha – akkor*” kapcsolaton alapuló korlátozást jelent. „*Ha*” a ható A entitás adott működésű, „*akkor*” a hatás nem engedélyezi a függő B entitás (más esetben lehetséges) bizonyos működéseit, így ebben az esetben az nem tetszőleges működésű, azaz B -re a tartományánál erősebb korlátozó adható meg. Az A hatása B -re olyan elempárokként adható meg, mely első eleme az A adott működé-

sei, második eleme pedig a B működésének korlátozója. Ha az A az első elemnek megfelelő működésű, akkor a B működése korlátozható az elempár második elemével.

A *ható* entitás hatása a *függő* entitás működésének *módosulását* eredményezi, egyetlen konkrét működés vizsgálatával ezért nem deríthetők fel a részek közötti hatások. Egyrészt tehát a függő entitás több működését kell vizsgálnunk, mivel így állapítható meg a változásának a tartománya. Másrészt, általában nem a ható entitás egyetlen működésének, hanem adott *jelleghű* működésének, azaz működéscsoportjának a hatását szeretnénk vizsgálni. Ehhez az szükséges, hogy a függő entitás tartományának megállapításához kiválasztott működések között szerepeljenek az adott jellegű működések, amit úgy nevezünk, hogy a ható entitás működései *értelmezhetők* a vizsgált működések esetén.

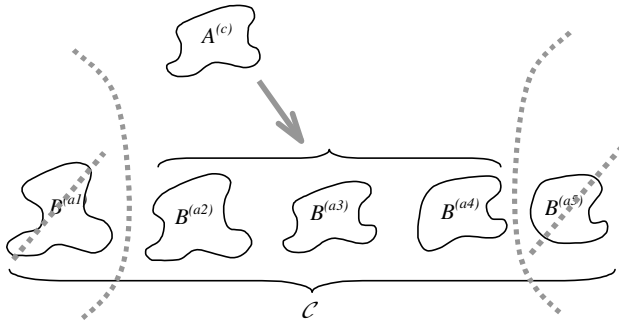
Az értelmezhetőség a feltételargumentumoknak egy részhalmazát választja ki, azokat a működéseket, amelyek esetén az entitás a működéscsoport valamelyik tagjának megfelelően változik. Hasonlóan, egy entitás egy c működésének adott C feltételű értelmezése a C azon részhalmazát jelenti, amelyben az entitás pontosan a c működésének megfelelően változik. Ha a feltételargumentumok ezen részhalmaza üres, akkor az entitásnak ez a működése nem értelmezhető az adott feltételek mellett.

▼ **Működés és működés adott feltételű értelmezhetősége.** A $\sigma^{(A)}$ kompozitban az A entitás $A^{(C')}$ működéseinek $C \subseteq \mathcal{A}$ feltételű értelmezése a $C_{|A^{(C')}}$ módon jelölt $\{c \in C \mid \exists c' \in C' : A^{(c)} = A^{(c')}\}$ részhalmaza (a C azon argumentumai, melyek esetén az A működése megegyezik valamely $A^{(C')}$ működéssel). Az $A^{(C')}$ működések értelmezhetők a $C \subseteq \mathcal{A}$ feltétel esetén, ha $C_{|A^{(C')}} \neq \emptyset$. Egyetlen $A^{(c)}$ működés C feltételű értelmezését $C_{|A^{(c)}}$ ($C_{|A^{(\{c\})}}$) módon jelöljük, illetve egyetlen $A^{(c)}$ működést *értelmezhetőnek* nevezünk a C feltétel esetén, ha az $A^{(\{c\})}$ értelmezhető.

▼ **Konkrét működés hatása, függő entitás.** A $\sigma^{(A)}$ absztrakt megadású kompozitban az A entitás $C \subseteq \mathcal{A}$ feltételnél értelmezhető $A^{(c)}$ konkrét működésének hatása a $T' \subseteq T$ időpontokban a B entitás $B^{(C)}$ működéseire a

B entitás $B_{(\leftarrow_{T'}^{(C)} A^{(c)})}$ módon jelölt $\bigcup B_{|_{T'}^{(C)} A^{(c)}} = \bigcup B_{|_{T'}^{(\{a \in C | A^{(c)} = A^{(a)}\})}}$ absztrakt működése. Az A entitás $A^{(c)}$ (konkrét) működése a T' időpontokban *hat* a *függő* entitás $B^{(C)}$ működéseire (jelölés: $A^{(c)} \rightarrow_{T'}^{(C)} B$, ellenkező esetben: $A^{(c)} \nrightarrow_{T'}^{(C)} B$), ha $B_{(\leftarrow_{T'}^{(C)} A^{(c)})} \subset \bigcup B_{|_{T'}^{(C)}}$ (hatása az időpontokban korlátozza a B működését).

Egy entitás egyetlen (a feltételek mellett értelmezhető) működésének a hatását valamely más entításra tehát úgy definiáljuk, hogy a ható entitás adott működésének megfelelő esetekben a függő entitás a működéseinek tartományánál kevesebb állapotlehetőségen belül változhat. A függő entitás számára „engedélyezett” működések együttese, azaz az absztrakt működés a ható entitás hatása. Ez a „hatás” megegyezhet a cél-entitás működésének a tartományával is, ami azt jelenti, hogy a kiindulási entitás ténylegesen nem hat a cél-entításra, így ez utóbbi a tartománya által megszabott összes lehetőségnek megfelelően is változhat.



25. ábra. A hatás adott „mederbe tereli” a függő entitás változását.

▽ Működések hatása. A $\sigma^{(A)}$ absztrakt megadású kompozitban az A entitás $C \subseteq \mathcal{A}$ feltételnél értelmezhető $A^{(C')}$ működéseinek hatása a $T' \subseteq \mathcal{T}$ időpontokban a B entitás $B^{(C)}$ működéseire a B entitás $B_{(\leftarrow_{T'}^{(C)} A^{(C')})}$ módon jelölt $\bigcup B_{|_{T'}^{(C)} A^{(C')}} = \bigcup B_{|_{T'}^{(\{a \in C | \exists c \in C': A^{(c)} = A^{(a)}\})}}$ absztrakt működése. Az $A^{(C')}$

működések a T' időpontokban *hatnak* a *függő* entitás $B^{(C)}$ működéseire (jelölés: $A^{(C')} \rightarrow_{T'}^{(C)} B$, ellenkező esetben: $A^{(C')} \nrightarrow_{T'}^{(C)} B$), ha $B_{(\leftarrow_{T'}^{(C)} A^{(C')})} \subset \bigcup B_{T'}^{(C)}$.

§ Működések hatásának számítása: $B_{(\leftarrow_{T'}^{(C)} A^{(C')})} = \bigcup_{C \in C'} B_{(\leftarrow_{T'}^{(C)} A^{(C)})}$, azaz az A entitás C' működéseinek a hatása az egyes működések hatásának uniójaként jelenik meg.

Egy entitás több működése együttesen a működések hasonlítható absztrakciójaként, azaz következményeként jelenik meg. Egyetlen következmény így alkalmas több működés megadására is.

▽ Következmény működéseinek hatása. A $\sigma^{(\mathcal{A})}$ absztrakt megadású kompozitban az A entitás bizonyos működéseinek A' következménye egy vagy több, $A^{(C_{A'})}$ működést határoz meg. Az A entitás A' következménye által meghatározott, a $C \subseteq \mathcal{A}$ feltételnél értelmezhető *működéseinek hatása* a $T' \subseteq \mathcal{T}$ időpontokban a B entitás $B^{(C)}$ működéseire a B entitás $B_{(\leftarrow_{T'}^{(C)} A')}$ módon jelölt $\bigcup B_{T'}^{\{a \in \mathcal{A} \mid A' \Leftarrow A^{(a)}\}}$ absztrakt működése. Az A entitás A' következménye által meghatározott *működések* a T' időpontokban *hatnak* a *függő* entitás $B^{(C)}$ működéseire (jelölés: $A' \rightarrow_{T'}^{(C)} B$, ellenkező esetben: $A' \nrightarrow_{T'}^{(C)} B$), ha $B_{(\leftarrow_{T'}^{(C)} A')} \subset \bigcup B_{T'}^{(C)}$.

§ Következmény által meghatározott működések hatásának számítása: $B_{(\leftarrow_{T'}^{(C)} A')} = \bigcup_{C \in C'} B_{(\leftarrow_{T'}^{(C)} A^{(C_{A'})})}$, azaz ekkor azokra a működésekre kell elvégeznünk az uniót, amelyek teljesítik a következmény feltételét.

Az A entitásnak a B -re azonos hatást kifejtő különböző konkrét működesei egyetlen következményként adhatók meg, így az egyes konkrét működések hatásai összevonhatók.

§ A feltétel bővülésével el is veszhet, de meg is jelenhet egy hatás. Vizsgáljuk meg, hogy a feltételek bővülésével hogyan változik egy $A^{(c)}$

működés $T' \subseteq T$ időpontokban fennálló $B_{(\leftarrow_{T'}^{(C)}) A^{(c)}} = \bigcup B_{T'}^{(\{a \in C \mid A^{(c)} = A^{(a)}\})}$ hatása. A $C' \subseteq C$ esetén a $T' \subseteq T$ időpontokban függő $B^{(C')}$ működések $\bigcup B_{T'}^{(C')}$ tartományánál legfeljebb bővebb lehet az $\bigcup B_{T'}^{(C)}$ tartomány. Ugyanakkor a bővebb C feltételben lehet olyan c'' , mely esetén az $A^{(c'')} = A^{(c)}$, így az $A^{(c)}$ működés $\bigcup B_{T'}^{(\{a \in C \mid A^{(c)} = A^{(a)}\})}$ hatása is lehet bővebb (de szűkebb nem). A feltétel bővülésével így el is veszhet, de meg is jelenhet egy hatás.

▽ Hatás és függés. A $\sigma^{(A)}$ kompozitban az A entitás a $T' \subseteq T$ időpontokban *hat* a B entitás $B^{(C)}$ működéseire (jelölés: $A \rightarrow_{T'}^{(C)} B$ ellenkező esetben: $A \not\rightarrow_{T'}^{(C)} B$), illetve a $B^{(C)}$ működések *függnek* az A entitástól (jelölés: $B \leftarrow_{T'}^{(C)} A$, ellenkező esetben: $B \not\leftarrow_{T'}^{(C)} A$), ha létezik az A olyan C feltételek mellett értelmezhető működése, amely az időpontokban *hat* a $B^{(C)}$ működésekre. A hatás lehet temporális és strukturális is.

Egy entitás tehát akkor *hat* egy másik entitásra, ha valamely működései a függő entitást a tartománykorlátozójánál erősebben korlátozzák, azaz annak működésére a tartományánál szűkebb lehetőségeket határoznak meg.

A hatás a ható entitás adott működései, azaz adott feltételargumentumok esetén következik be. Így azt is vizsgálhatjuk, hogy egy kiválasztott entitás egy cél-entitásra milyen feltételargumentumok mellett *hat*. A feltételargumentumok üres-halmaza esetén nem létezik ilyen hatás, egyébként igen, ill. a teljes argumentumhalmaz esetén ez a hatás feltétlen.

▽ Hatás feltétele. A $\sigma^{(A)}$ kompozitban az A entitás által a $T' \subseteq T$ időpontokban a B entitás $B^{(C)}$ működéseire kifejtett hatásának feltétele az $\mathcal{A}$ argumentumhalmaz $C_{A \rightarrow_{T'}^{(C)} B}$ módon jelölt $\{c \in \mathcal{A} \mid A^{(c)} \rightarrow_{T'}^{(C)} B\}$ részhalmaza (mely a $\sigma^{(A)}$ feltétele).

☞ $A \mathcal{C}_{A \rightarrow_{T'}^{(C)} B} = \emptyset$ pontosan akkor, ha az időpontokban az A nem hat a $B^{(C)}$ működésekre.

☞ Az időpontok bővebb halmaza esetén legfeljebb bővíülhet egy entitás hatásának feltételhalmaza: $T'' \subseteq T' \subseteq T$ esetén $\mathcal{C}_{A \rightarrow_{T''}^{(C)} B} \subseteq \mathcal{C}_{A \rightarrow_{T'}^{(C)} B}$.

☞ Strukturális hatás feltétele: $T' \subseteq T$ esetén $\mathcal{C}_{A \rightarrow_{T'}^{(C)} B} \subseteq \mathcal{C}_{A \rightarrow^{(C)} B}$.

▽ Kölcsönhatás, kapcsolat és függetlenség. A $\sigma^{(A)}$ kompozit $T' \subseteq T$ időpontjaiban a $\mathcal{C}$ feltételnél fennálló kölcsönös hatást ($A \leftarrow_{T'}^{(C)} B$ és $A \rightarrow_{T'}^{(C)} B$) *kölcsönhatásnak* nevezzük (jelölés: $A \leftrightarrow_{T'}^{(C)} B$, strukturális esetben: $A \leftrightarrow^{(C)} B$). Ha a hatás irányától eltekintve az egyik entitás hat a másikra ($A \leftarrow_{T'}^{(C)} B$ vagy $A \rightarrow_{T'}^{(C)} B$), akkor a két entitás *kapcsolatban* van egymással (jelölés: $A \sim_{T'}^{(C)} B$, strukturális kapcsolat esetén $A \sim^{(C)} B$). Ha a hatás az egyik irányban sem áll fenn, akkor az időpontokra vonatkozóan a két entitás *független* (jelölések: $A \nmid_{T'}^{(C)} B$, strukturális esetben: $A \nmid^{(C)} B$).

▽ Invariancia. A strukturális függetlenséget ($A \nmid^{(C)} B$) invarianciának nevezzük.

$A \rightarrow B$	Hatás
$A \leftarrow B$	Függés
$A \leftrightarrow B$	kölcsönhatás
$A \sim B$	kapcsolat
$A \nmid B$	függetlenség

A hatással kapcsolatos fogalmak jelölése

☞ *Kölcsönös hatások.* A $\mathcal{C}_1$ és $\mathcal{C}_2$ nem diszjunkt feltételek, valamint $A \leftarrow_{T'}^{(\mathcal{C}_1)} B$ és $A \rightarrow_{T'}^{(\mathcal{C}_2)} B$ esetén $A \leftrightarrow_{T'}^{(\mathcal{C}_1 \cap \mathcal{C}_2)} B$, azaz közös feltételargumentumok esetén a két entitás között kölcsönhatás van.

☞ *Hatások és kapcsolat.* A $\mathcal{C}_1$ és $\mathcal{C}_2$ feltételek, valamint $A \leftarrow_{T'}^{(\mathcal{C}_1)} B$ és $A \rightarrow_{T'}^{(\mathcal{C}_2)} B$ esetén $A \sim_{T'}^{(\mathcal{C}_1 \cup \mathcal{C}_2)} B$.

✓ **Változó és változatlan működésű rész.** Absztrakt megadású rendszer $\sigma^{(C)}$ működései esetén az E entitás (rész) változó, ha létezik legalább két eltérő működése, ellenkező esetben változatlan.

☞ *Az egyetlen működés változatlan működés.*

☞ *Adott feltételargumentumok esetén a tetszőleges absztrakt működésű entitás változatlan működésű.*

§ **Változatlan működésű entitás nem hat entitásra és nem függ entitástól,** így önmagára sem hat, illetve önmagától sem függ. Ekkor az entitást csak maga a változás függvénye korlátozhatja.

☞ *Tetszőleges absztrakt rendszer egyetlen entitásra sem hat.*

§ **Példány entitása strukturálisan és értékkorlátozóként hat a változó részentitásaira.**

☞ *Példány változó entitása strukturálisan és értékkorlátozóként hat önmagára.*

§ **Tranzitív hatás.** A $\sigma^{(A)}$ kompozitban $A' \rightarrow^{(C)} B'$, $B'' \rightarrow^{(C)} C'$, valamint $B' \subseteq B''$ (a működések B'' következménye a B entitásnak a B' működéseit is meghatározza, pl.: bővebb feltételhalmaz, vagy egy absztraktabb következmény) esetén $A' \rightarrow^{(C)} C'$.

Az A bizonyos működéseinek hatása így a $\sigma^{(A)}$ több $B_{j \in J}$ entitásának C_j működéseire is kiterjedhet.

§ **A kapcsolatok nem feltétlenül tranzitívak.** Például $A \rightarrow B$ és $A \rightarrow C$ esetén B és C független is lehet.

Hatások erőssége

A hatás maga is (közvetett) korlátozásként jelenik meg, így korábbi definíciónk alapján beszélhetünk a *hatások mint korlátozók* adott szempont szerinti gyengességéről, illetve erősségéről. A hatás mint korlátozó funkcionális

korlátozó esetén a legerősebb, illetve maximális gyengességű, ha nem áll fenn a hatás.

időlegesség	„erősség”	irányultság
strukturális	funkcionális	hatás/függés
időleges	relációs	kapcsolat

Hatások (függések) tipizálása

§ Az absztrakció gyengíti a hatásokat mint korlátozókat. A működések hasonlítható absztrakciójára gyengébb következmények fogalmazhatók meg, így azokra gyengébb hatások mint korlátozók adhatók meg.

A hatások között egy más jellegű gyengesség/erősség is definiálható. Mivel az absztraktabb működésre gyengébb korlátozók adhatók meg, ezért a rendszer egymást követő absztrakciói során „elvesznek” a hatások, így növekednek a függetlenségek, illetve invarianciák. Adott szempont szerint erősebbnek tekinthetünk egy hatást, illetve kapcsolatot, ha az a szempont nagyobb értékű absztrakciójánál tűnik el.

§ Hatás nem feltétlenül teljesül a működések absztrakciójára. Vizsgáljuk meg egy $\sigma^{(A)}$ kompozit $A^{(C')} \rightarrow_T^{(C)} B$ hatását, ha a $\sigma^{(A)}$ működéseket azok $\sigma'^{(A)}$ következményeire cseréljük. A hatás azt jelenti, hogy az A entitás bizonyos $A^{(C')}$ működései a $B^{(C)}$ működéseket azok tartományánál jobban korlátozza, azaz $B_{(\leftarrow_T^{(C)} A^{(C')})} \subset \bigcup B_{T'}^{(C)}$. A hatás „elveszhet”, azaz lehetséges, hogy az a működések $\sigma'^{(A)}$ hasonlítható absztrakciói esetén már nem áll fenn. Egyrészt a ható A entitás különböző $A^{(c_1)}$ és $A^{(c_2)}$ (C feltétel esetén értelmezhető) működései a működések hasonlítható absztrakciójában már megegyezhetnek, így például egy $A^{(c)}$ működés absztrakciója egy olyan működéssel válhat azonossá, mely nem hat a $B^{(C)}$ működésekre. Másrészt a működések absztrakcióját tekintve a $B^{(C)}$ működések is absztraktabbá válhatnak, így a hatás (mint a B entitás $B^{(C)}$ működéseinek a következménye) is gyengül.

§ Működések absztrakciója esetén megjelenhet hatás. Például a $\sigma^{(A)}$ rendszerben egy $A^{(c)} \rightarrow B$ (azaz $\bigcup B^{(\mathcal{C}_{|A^{(c)}})} = \bigcup B^{(C)}$) esetén a $\sigma^{(\mathcal{C}_{|A^{(c)}})}$ működésekét hagyjuk változatlanul, a $c' \in \mathcal{C} \setminus \mathcal{C}_{|A^{(c)}}$ argumentumú működéshez pedig rendeljük egy olyan következményét, melyben $B^{(c')}$ nem következménye az eredeti $\bigcup B^{(C)}$ tartománynak. Ekkor megjelenik az $A^{(c)}$ működés hatása a $B^{(c')}$ működésekre, mivel $\bigcup B'^{(\mathcal{C}_{|A^{(c)}})} = \bigcup B^{(C)} \subset \bigcup B'^{(C)}$.

Működések hasonlítható absztrakciói esetén tehát lehetséges, hogy bizonyos hatások már nem teljesülnek. Ugyanakkor az absztrakció során megjelenhetnek olyan „látszólagos” hatások is, melyek nem teljesültek az eredeti rendszerben. Egy ilyen „látszólagos” hatás megjelenését az okozza, hogy elvesztjük a függő entitás tartományára vonatkozó információkat, mivel feltételeztünk az eredeti tartományon kívüli működést is. Ekkor hatást kifejtőként jelenik meg az a működés, mely a függő entitást csak az eredeti tartományra korlátozza.

▽ Működések következményének látszólagos hatása. A $\sigma^{(A)}$ működéseinek $\sigma'^{(A)}$ hasonlítható absztrakcióiban fennálló hatást a $\sigma^{(A)}$ *következményében megjelenő látszólagos hatásnak* nevezzük, ha az nem áll fenn a $\sigma^{(A)}$ működések esetén.

Ha egy rendszer működéseit csak egy adott absztrakciós szintig ismerjük, akkor lehetségesnek tarthatjuk egy entitás adott feltételek melletti tényleges tartományán kívüli működését. A látszólagos hatások így a tartományok tisztázásával (mely a működések hasonlítható pontosítása) küszöbölhetők ki.

▽ Működések tartományabsztrakciója. Absztrakt megadású rendszer $\sigma^{(C)}$ működéseinek $\sigma'^{(C)}$ hasonlítható absztrakcióit a működések tartományabsztrakciónak nevezzük, ha minden E entitás esetén a működések tartománya azonos marad ($\bigcup E^{(C)} = \bigcup E'^{(C)}$).

§ **Tartományabsztrakció esetén nem jelenhet meg új hatás.** Tegyük fel, hogy a $\sigma^{(C)}$ működések tartományabsztrakciójaként kapott $\sigma'^{(C)}$ működések esetén a rendszerben létezik olyan $A \rightarrow_{T'}^{(C)} B$ hatás, amely nem volt jelen a $\sigma^{(C)}$ -ben. Ekkor létezik legalább egy olyan c argumentum, hogy $\sigma'^{(C)}$ -ben $A^{(c)} \rightarrow_{T'}^{(C)} B$, $\sigma^{(C)}$ -ben viszont $A^{(c)} \nrightarrow_{T'}^{(C)} B$, azaz $\bigcup B^{(C_{|A^{(c)}})} = \bigcup B^{(C)}$ (a B entitás $C_{|A^{(c)}}$ működéseinek uniói megadják a tartományt). A működések hasonlítható absztrakció során az $\bigcup B^{(C_{|A^{(c)}})}$ hatás legfeljebb bővílhet. Mivel tartományabsztrakciót alkalmaztunk, ezért a $\bigcup B^{(C)}$ tartománya nem változhat (nem bővílhet), illetve azt a $\bigcup B^{(C_{|A^{(c)}})}$ hatás sem haladhatja túl. Ezért a $\sigma'^{(C)}$ -ben az $A^{(c)}$ működés B -re kifejtett $A^{(c)} \rightarrow_{T'}^{(C)} B$ hatása megegyezik az $\bigcup B^{(C)}$ tartománnyal, így a $\sigma'^{(C)}$ -ben az A nem fejt ki hatást a B -re, azaz ellentmondásra jutottunk.

§ **Tartományabsztrakció növeli a függetlenségeket, illetve invariánciákat.**

▽ **Hatások gyengesége.** A $\sigma^{(A)}$ kompozit $A^{(C')} \rightarrow_{T'}^{(C)} B$ hatásának a $\sigma^{(A)}$ működéseire rögzített $\prec$ szempont szerinti gyengesége az a legnagyobb r érték, mely esetén az $A^{(C')} \rightarrow_{T'}^{(C)} B$ hatás még fennáll a $\prec_{(r)}(\sigma^{(A)})$ rendszerben.

▽ **Kölcsönhatás és kapcsolat gyengesége.** Egy $\sigma^{(A)}$ kompozit A és B entitásai között fennálló kölcsönhatás, illetve kapcsolat adott szempont szerinti gyengesége az a legnagyobb érték, mely esetén az absztrakt rendszerben a kölcsönhatás, illetve a kapcsolat még fennáll.

Entitás, környezete és külvilága

A rendszerből kiemelhetjük, és vizsgálatunk tárgyául választhatjuk annak egy részletét, egy entitását. Egy kompozit univerzumként szolgáló teljes rendszere így az *entitásra* és az azt körülvevő világra bontható a

$\sigma = E \otimes E_{(\emptyset)}$ módon. Az $E_{(\emptyset)}$ komponensből elhagyhatjuk azokat az $E_{(+)}$ komponenseket – a *külvilágot* –, amelyek sem nem határozzák meg E -t, sem E nem hat rájuk. Az $E_{(\emptyset)}$ megmaradó $E_{(-)}$ részét a rendszer *környezetének* nevezzük.

✓ **Entitás bemenete, kimenete és transzfere.** A $\sigma^{(A)}$ rendszer E entitása esetén a rendszer entitáson kívüli azon részeit együttesen, amelyek hatnak az entitásra az entitás *bemenetének* (jelölés: $E_{(\leftarrow)}$), amelyek függenek az entitástól az entitás *kimenetének* (jelölés: $E_{(\rightarrow)}$) nevezzük. Az entitás *transzferét* (jelölés: $E_{(\leftrightarrow)}$) a rendszer azon részei alkotják, amelyek egyszerre hatnak és függenek is az entitástól, azaz a bemenetében és kimenetében is szerepelnek.

✓ **Entitás környezete és külvilága.** Rendszer entitásának a *környezete* (jelölés: $E_{(-)}$) a bemenetet és kimenetet alkotó részek együttesen, tehát a rendszer entitáson kívüli azon részei, amelyek hatnak az entitásra vagy attól függenek. A rendszer entitáson és környezetén kívüli részei alkotják az entitás *külvilágát* (jelölés: $E_{(+)}$), amely részek sem nem hatnak sem nem függenek az entitástól, tehát nincsenek az entitással kapcsolatban.

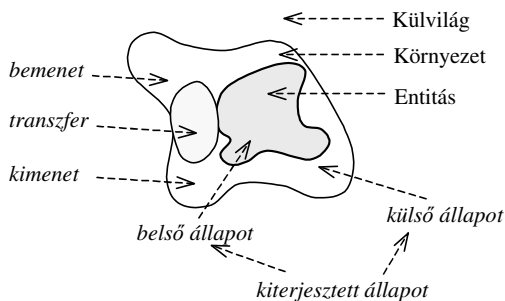
☞ *Entitás bemenete, kimenete, transzfere, környezete és külvilága egyaránt tekinthetők entitásoknak.*

✓ **Kiterjesztett entitás.** Entitást és környezetét együttesen kiterjesztett entitásnak nevezzük.

§ **Két entitás pontosan akkor független, ha egyik sincs benne a másik környezetében,** azaz egyik sem a környezetével együtt vett másik entitás részentitása.

✓ **Entitás belső, külső és kiterjesztett változása és állapota.** A rendszer egy működésének az E entitásra leszűkített $E^{(a)} = \sigma_{|E}^{(a)}$ változását az entitás *belső változásának*, adott időpontban felvett $E^{(a)}(t) \subseteq S_{|E}$ értékét

pedig az entitás *belső állapotának* nevezzük. Hasonlóan, a környezetre leszűkített $E_{(\sim)}^{(a)}$ változás, illetve állapot a *külső változás*, illetve a *külső állapot*. Az entításra és környezetre leszűkített változást, illetve állapotot pedig az entitás *kiterjesztett változásának*, illetve *kiterjesztett állapotának* nevezzük.



26. ábra. Entitás (komponens) és környezete.

§ **Entitás kiterjesztett és belső változása a működések modellje**, mivel a működések leszűkítése egy egyszerű absztrakció, így egyben típus-absztrakció.

§ **Entitás kiterjesztett/belső változása a működések következménye**, így nem tetszőleges absztrakt változások esetén a működésekre vonatkozó információ is.

§ **Entitás meghatározza a működések kettős absztrakcióját**. A $\sigma^{(C)}$ működések első $\alpha_{(+)}$ absztrakciója elvonatkoztat az entitás külvilágától, az entitás $\alpha_{(+)}$ $\sigma^{(C)}$ kiterjesztett változásának $\alpha_{(\sim)}$ absztrakciója pedig elvonatkoztat az entitás környezetétől, így $E^{(C)} = \alpha_{(\sim)}(\alpha_{(+)}(\sigma^{(C)}))$.

§ **Két egymásutáni tartományabsztrakció meghatározza a rendszer egy entitását**. A működések (hasonlítható) tartományabsztrakciói esetén az eredeti rendszerben fennálló hatások „elveszhetnek”, azaz már nem feltétlenül állnak fenn az absztraktabb rendszerben. A $\sigma^{(C)}$ kompozit működéseink hajtsuk végre az $\alpha_{(+)}$ tartomány absztrakciót, majd ezeken a

$\sigma'^{(C)} = \alpha_{(+)}(\sigma^{(C)})$ következményeken az $\alpha_{(\sim)}$ módszerű,
 $\sigma''^{(C)} = \alpha_{(\sim)}(\sigma'^{(C)}) = \alpha_{(\sim)}(\alpha_{(+)}(\sigma^{(C)}))$ eredményű tartomány absztrakciót.

Rögzítsük a rendszer helyeinek $L' \subseteq L$ részhalmazát. A $\sigma'^{(C)}$ működések esetén is megmaradó hatások alapján határozzuk meg az L' helyeken elhelyezkedő entitás $\sigma'_{L'}^{(C)}$ működéseivel kapcsolatban lévő L'' helyeket, illetve a $\sigma''^{(C)}$ működések esetén is megmaradó hatások alapján a $\sigma''_{L'}^{(C)}$ működéseivel kapcsolatban lévő L''' helyeket ($L''' \subseteq L''$). Az L' helyek rögzítése esetén a kettős absztrakció meghatározza a rendszer egy, az $L' \cup L'''$ helyeken elhelyezkedő E entitását, illetve annak $E^{(C)} = \sigma_{(L' \cup L''')}^{(C)}$ működését, valamint az E entitás környezetét és a $\sigma_{(L' \cup L' \cup L''')}^{(C)}$ kiterjesztett változását.

▽ Entitás körvonalazása két egymásutáni tartomány absztrakcióval. A $\sigma^{(C)}$ működések esetén az $\alpha_{(+)}$ és $\alpha_{(\sim)}$ tartomány absztrakciók segítségével a $\sigma_{L'}^{(C)}$ ($L' \subseteq L$) részentitású entitás meghatározását az entitás körvonalazásának nevezzük.



Időben lezajló változás

Az időben lezajló változást a következőkben alapvetően egyetlen kiválasztott entitás szempontjából vizsgáljuk. Kiindulópontunk az entitás és környezetének egymásrahatása. A hatások szempontjából az entitás állapota, illetve annak megváltozása hathat környezetére (akció), illetve a környezet állapota a megváltozásával, azaz egy külső esemény következtében hathat az entitásra. A hatás felvétele az entitás transzformációjaként jelenik meg. Az entitás aktív módon is változhat, azaz tevékenységeket hajthat végre.

Objektumok rendszere

Előkészítésként először néhány kompozitként történő ábrázolást vizsgálunk. Az állítások segítségével egy bonyolultabb felépítésű rendszer ekvivalens módon egyszerűbb rendszerré alakítható, így az egzaktul, mégis egyszerűbben ábrázolható. Másrészt bizonyos állításokat elegendő az egyszerűbb formára megfogalmazni, mivel a megfelelő átjelölés megfordításával azok bonyolultabb rendszerekre is alkalmazhatók.

Egy kompozit rendszer részei az egyes $l \in L$ helyekhez kötöttek. A rendszerben természetesen lehetnek olyan objektumok, amelyek „átúsznak” a rendszeren, azaz változtatják elhelyezkedésüket, vagy az időben változva több vagy kevesebb helyet foglalnak el. Az első állítások arra vonatkoznak, hogy az objektumokat tartalmazó rendszer átjelölhető olyan rendszerré, amelyben az objektumok nem változtatják elhelyezkedésüket, így az objektumrendszerek vizsgálhatók egyszerűbb, helyhez kötött részekként, azaz objektumrendszer helyett elegendő az entitások rendszerét vizsgálnunk.

Az objektumok egyszerűsített képei az entitások. Az entitások a rendszer adott helyeihez kötöttek, illetve a helyeket kizárólagossággal foglalják el, azaz két entitás különböző időpontokban sem lehet azonos helyen.

✓ Objektum, objektum-azonosító. A $\sigma = (\mathcal{T}, L, \overline{\mathcal{T} \times L}, S, P)$ rendszerben $T_o \subseteq T$ és $\pi_o : T_o \rightarrow 2^L \setminus \{\emptyset\}$ esetén a σ rendszer T_o idejű (más néven: *élettartamú*) és π_o *objektum-azonosító* o objektuma a σ leszűkítése a $\{(t, l) \in \overline{\mathcal{T} \times L} \mid t \in T_o \wedge l \in \pi_o(t)\}$ pozíciókra. Az objektum a $t \in T_o$ időpontban a rendszer $\pi_o(t) \subseteq L$ helyeit foglalja el és $o(t) = \otimes_{l \in \pi_o(t)} P(t, l)$ állapotban van.

✓ Inicializálás, terminálás, végtelen életciklus. Ha egy σ rendszerben az o objektum T_o élettartamának létezik minimuma, azt az o *inicializálásának*, ha létezik maximuma, azt az o *terminálásának* nevezzük. Ha az o élettartamának nem létezik maximuma, akkor az o egy *végtelen életciklusú* objektum.

✓ Részobjektum. Az o_A objektum *részobjektuma* az o_B , ha $T_{o_A} \supseteq T_{o_B}$, valamint $\forall t \in T_{o_B} : \pi_{o_A}(t) \subseteq \pi_{o_B}(t)$.

☞ Minden objektum önmaga (triviális) részobjektuma.

✓ Rendszer objektum-rendszerként megadása. Az $\otimes^{(\pi_{o_i})}_{o_i \in I}$ a $\sigma = (\mathcal{T}, L, \overline{\mathcal{T} \times L}, S, P)$ rendszer objektum-rendszerként történő megadása, ha o_i a σ objektumai, valamint minden időpont esetén az objektumazonosítók együttesen megadják a rendszer értelmezett pozícióit, azaz: $\overline{\mathcal{T} \times L} = \bigcup_{i \in I} \{(t, l) \in \overline{\mathcal{T} \times L} \mid t \in T_{o_i} \wedge l \in \pi_{o_i}(t)\}$.

§ Rendszer entitása a rendszer objektuma, így entitásokra is alkalmazhatók az *inicializálás*, *terminálás*, illetve a *végtelen életciklus* fogalmak. Az objektum-rendszerként történő megadás analógiájára az „entitás-rendszerként” való felírás nem igényel külön definíciót, mivel az egyszerűen a rendszer kompozícióként történő megadása.

§ Objektum-rendszer kompozícióvá átjelölhető. Az $\otimes^{(\pi_{o_i})}_{o_i \in I}$ objektum-rendszerként felírt rendszer átjelölhető olyan, vele azonos idejű rendszerré, melyben az objektumoknak az átjelölt rendszer entitásai feleltethetők meg kölcsönösen egyértelmű módon. Ilyen átjelölt rendszer például a

$\sigma = (\mathcal{T}, L, \overline{\mathcal{T} \times L}, \otimes_{i \in I} S_{o_i}, P)$ rendszer, ahol minden S_{o_i} azonos a kiindulási rendszer állapothalmazával, valamint $P(t, i) = o_i(t)$.

§ **Összetett állapotú komponens átjelölése kompozícióvá.** Ha egy $\sigma = (\mathcal{T}, L, \overline{\mathcal{T} \times L}, \otimes_{l \in L} S_l, P)$ kompozit egy l helyének megfelelő S_l állapothalmaza összetett, azaz maga is előáll kettő vagy több alaphalmaz általánosított Descartes-szorzataként ($S_l = \otimes_{i \in I} S_{l_i}$), akkor a rendszer átjelölhető egy, a σ rendszerrel azonos idejű $\sigma' = (\mathcal{T}, L', \overline{\mathcal{T} \times L'}, \otimes_{l \in L'} S_l, P')$ rendszerre, ahol $L' = (L \setminus \{l\}) \cup I$ (ha szükséges, I elemeit átjelöljük, hogy különbözzenek L elemeitől). A P' függvénynek az $L \setminus \{l\}$ helyeken az eredeti P -vel azonos állapotkombinációkat kell megadni, az „új” I helyeken pedig az eredeti összetett állapotkombinációk megfelelő elemét ($P'(t, i) = P(t, l_i)$).

§ **Összetett állapotkomponenseket tartalmazó rendszer átjelölhető eleminek tekintett állapotú részek kompozíciójává,** mivel az állítás tetszőleges sok összetett állapotkomponensre is kiterjeszthető.

§ **Adott részeket kiválasztva a rendszer átjelölhető olyan rendszerre, melyben az adott részeknek egyetlen, összetett állapotú komponens felel meg,** mivel az átalakítás ellentétes irányban is elvégezhető.

Objektum-rendszer így átalakítható olyan kompozícióvá, ahol az eredeti objektumoknak az átjelölt rendszer entitásai felelnek meg, valamint a rendszer minden egyes $l \in L$ helyén csak eleminek tekintett (további részeire nem bontott) állapotok kombinációját veszi fel.

Esemény és működés

Egy rendszer változásának egyetlen időtartam kezdő és zárópontjára való leszűkítése a rendszer kvantálása. Ez a változás legkisebb egysége, a változás kvantuma, amit a korábbi definíciónk alapján *állapotváltásnak* (kvantumváltásnak) vagy lépésnek nevezünk.

Az állapotváltás „korábbi” időpontját *bázis-* vagy *viszonyítási időpontnak*, a „későbbi” (általában az aktuálisnak tekintett) időpontot pedig a *vizsgált időpontnak*. Az állapotváltás a viszonyítási időponthoz képest vett esetleges változást adja meg a vizsgált („aktuális”) időpontra vonatkozóan.

Az állapotváltás esetenként lehet identikus is, azaz a bázis és a vizsgált időpontban az állapot megegyezik, azaz nem történt tényleges állapotváltás. A tényleges állapotváltást *eseménynek* nevezzük, amelynek így van *ideje* (a vizsgálat időpontja), valamint kompozit rendszer esetén *forrása*, amely a megváltozott komponens.

Az állapotváltást a bázis és a vizsgált időpontok és az ott felvett állapotok $((t, s), (t', s'))$ elemkettesével adhatjuk meg. Amennyiben a két szituáció időpontjaitól, a „mikor”-tól is eltekintünk, akkor – determinisztikus megadás esetén – a kezdő és a záróállapot (s, s') kettősét kapjuk, amit *állapotátmenetnek*, vagy röviden *átmenetnek* nevezünk. Rendszer, vagy annak egy komponensének állapotátmenete egy kétidőpontú (kvantum) rendszer, amit egy *pozícionálással*, azaz az időpontok és a helyek megadásával vonatkoztathatunk a vizsgált rendszerre.

Átmenetként az állapotváltások a kezdő és záró időpontjuktól, kompozit esetén pedig az entitástól függetlenül adhatók meg. Egy átmenet így a különböző időpontokra (és entitásokra) alkalmazható állapotváltások halmazát, azaz absztrakt állapotváltást határoz meg.

Az eseményeket és a működést a következőkben a *determinisztikus megadási* móddal vizsgáljuk.

✓ **Esemény.** A nem identikus állapotváltást *eseménynek* nevezzük. Az eseménynek van *ideje* (a vizsgált „későbbi” időpont), illetve – kompozit esetén – *forrása*, amely a megváltozott entitás.

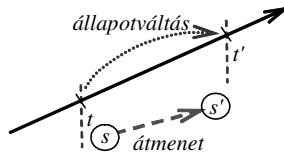
✓ **Átmenet, kiindulási- vagy forrás-, és célállapot.** Egy S állapotalmaz *átmenete* egy $(s, s') \in S \times S$ elempár, amelyet egy kétidőpontú (kvantum-) rendszernek tekintünk, ahol az s -et *kiindulási- vagy forrásállapotnak*, az s' -t *célállapotnak* nevezzük. Az átmenet ugyancsak jelölhető az $s \mapsto s'$ módon.

☞ Az állapotváltás megadható állapotátmenettel a megfelelő pozícionálás alkalmazásával (azaz a forráshelyek és időpontok megadásával).

✓ **Üres vagy identikus átmenet.** Az (s, s) egy *üres vagy identikus átmenet*, azaz ahol a forrás és a célállapot megegyezik.

Egy startpontú kvantumrendszer változása az egymást követő időpontokban egymásba kapcsolódó átmenetekként jelenik meg, ahol minden célállapot a következő esemény forrásállapota.

▼ **Konkrét átmenet, absztrakt átmenet.** A $\sigma = (\mathcal{T}, \mathcal{T}, S, P)$ rendszer *konkrét (állapot-)átmenete* az S állapothalmaz átmenete, *absztrakt átmenete* az absztrakt állapotok $2^S \setminus \{\emptyset\}$ átmenete. Az *absztrakt átmenet* állapothalmazhoz állapothalmazt rendel, így az *megadható konkrét átmenetek halmazaként*, azaz $S \times S$ nem üres részhalmazaként.



27. ábra. Állapotváltás és átmenet.

▼ **Állaputmódosító vagy állapot-transzformáció.** Egy S állapothalmaz *módosítója* vagy *transzformációja* egy $m: S \rightarrow S$ leképezés.

§ **Az állapot-transzformáció absztrakt állapotátmenetet határoz meg.** Az S állapothalmaz $m: S \rightarrow S$ leképezése olyan elemkettesek halmaza, amely egyértelmű, azaz minden forrásállapothoz csak egy célállapotot határoz meg (a célállapot azonban több forrásállapotból is elérhető).

Transzformáció: átmenet és akció

Szemléletünkben a rendszernek, illetve annak adott részének, entitásának is egyetlen jellemzője az adott időpontokban felvett állapota – a rendszerről nincs más információnk. A rendszer, illetve egy entitás változása így az állapot megváltozását jelenti.

A rendszer teljes változását egy időtartam kezdő és záró időpontjára leszűkítve a változás egy egyszerű állapotváltásként jelenik meg, illetve ha a pontos időpontoktól is eltekintünk, akkor egyetlen átmenetként.

Korábban megállapítottuk, hogy az absztrakt megadású rendszerek változásai átírhatók az átmeneti függvények formájába. Az átmeneti függvény egy rögzített időponthoz viszonyítva adja meg a változást.

§ Átmeneti függvénnyel megadott startpontú kvantumrendszer átírható tiszta átmeneti függvény formájába. Az átmeneti függvény formájában megadott startpontú kvantumrendszerek egy kitüntetett t_0 időpillanatban felvett állapotkombinációval parametrizálják az absztrakt megadású rendszert. Az átmeneti függvény a t_0 -ban felvett s_0 állapotkombinációhoz, valamint a t időponthoz rendeli hozzá a $s_t^{(s_0)} = P^{(S_0)}(s_0)(t)$ aktuális állapotkombinációt. Először vizsgáljuk azt az esetet, amennyiben a t_0 időpont a kvantumrendszerünk startpontja! Amennyiben a rendszert (egyszerű pontosítással) kiegészítjük az s_0 állapotkombinációval, mint résszel, valamint egy olyan résszel, amely minden t időpillanat esetén megadja a rákövetkező időpillanat t' értékét (amennyiben az létezik; egyébként definiálatlan), akkor az átmeneti függvény átírható tiszta átmeneti függvény formájára, amely mindössze az aktuális időpillanatban felvett állapotkombináció alapján adja meg a következő időpillanat állapotkombinációját, amennyiben az létezik (azaz nem az utolsó időpont). Amennyiben a t_0 nem a startpont, akkor a rendszert egészítsük ki egy olyan további résszel, amely értéke minden t időpontban az azt megelőző időpont.

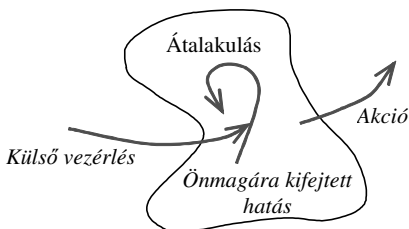
☞ Az állítás kiterjeszthető tetszőleges kvantumrendszerre, azaz: kvantumrendszer megadható tiszta átmeneti függvény formájában.

Kvantumrendszer változása tehát átírható a „tiszta” (reguláris-) átmeneti függvények formájába, ahol az átmeneti függvényre egyetlen feltételünk, hogy adott időpont esetén mindössze a rendszer állapotának ismeretében adja meg a következő (ill. esetlegesen előző) időpontban felvett állapotot, amennyiben az időpontban még értelmezett a rendszer. A tiszta átmeneti függvénnyel megadott (kvantum-) rendszer változása átmenetekként jelenik meg, a „tiszta átmeneti függvény” pedig állapot-transzformációként, azaz $m: S \rightarrow S$ leképezésként, amely az adott időpontban felvett s állapot esetén megadja a következő időpontban felvett s' állapotot, az „állapot rákövetkezőjét”.

§ **Startpontú kvantumrendszer megadható a startpontban felvett (determinisztikus esetben:) állapottal és a megfelelő állapot-transzformációval**, azaz egy állapot-transzformáció egy adott minimumú kvantált időhalmazra vonatkozóan meghatároz egy $P^{(\rightarrow)}$ tiszta átmeneti függvényt.

A tiszta átmeneti függvény olyan kényszerként (korlátozóként) is tekinthető, amely a kezdő időpillanatban felvett állapot alapján a rendszert a záró időpontban felvett állapotba váltja. Ezen megadás esetén ezért minden egyes időpont felvett állapota tekinthető úgy, mint ami hat a későbbi időpontokhoz tartozó állapotokra.

Rendszer adott entitásának változását önmaga és környezete (bemenete) határozza meg, ugyanakkor a változása hatással lehet a környezetre (a kimenetre). Az entitás szempontjából vizsgálva az entitással kapcsolatos hatások alapján a teljes rendszer változása két komponensre bontható. Az egyik komponens a belső állapot változása, az *átalakulás* vagy belső átmenet, amelyet maga az entitás és annak környezete (pontosabban: a bemenete) határoz meg. A másik komponens a rész hatása a környezetre (pontosabban: a kimenetre), amelyet a (rész) hatásának vagy *akciójának* nevezünk.



28. ábra. Külső vezérlés, átmenet és akció.

Rendszer adott entitásának adott időpillanatban felvett állapota így hatással lehet környezetének későbbi állapotára (akció), illetve a bemenettel együtt hatással lehet későbbi belső állapotára (átmenet).

	Entitás állapotára hat		Entitás állapotára hat
	bemenet	entitás	kimenet
hatás	külső vezérlés	önmagára kifejtett hatás	akció
eredménye	<i>(belső)átmenet</i>		...

Entitásra kifejtett hatás és entitás hatása

Adott változások tiszta átmeneti függvényként történő megadása esetén az egyetlen eszközünk a megfelelő állapot-transzformációs függvény felírása, amely a kiindulási állapothoz a rákövetkező időpontban érvényes célállapotot határozza meg, azaz állapothoz állapotot rendel.

A tiszta átmeneti függvény az entitás szempontjából tekintve egy függvénykettőssé, kettős állapot-transzformációvá bontható. Mindkét függvény a bemenet és az entitás együttes állapotán van értelmezve. A kettős első függvénye az entitás (esetleges) átmenetét, a másik pedig a kimenten szükséges (esetleges) állapotváltást határozza meg.

A kettős mindkét eleme egy-egy állapot-transzformációval kapcsolatos. Az állapot-transzformáció (determinisztikus esetben) adott állapothoz adott állapotot határoz meg, azonban lehetséges, hogy több különböző állapothoz is ugyanazt az eredményállapotot rendel. Ez azt jelenti, hogy az állapot-transzformációnak elegendő csak az állapotok adott csoportjait, azaz *absztrakt állapotokat* vizsgálni, mivel azonos állapotváltás történik a csoportba tartozó bármely konkrét állapot fennállása esetén.

Az entitás szempontjából a bemenet változását elegendő mindössze az absztrakt állapotok erejéig vizsgálni. És hasonlóan: ugyancsak elegendő mindössze az entitás absztrakt állapotát figyelembe venni a kimentre tett esetleges hatás (azaz az akció) esetén.

Interakciók

Az entitásokra bontott rendszer, illetve az azon belüli hatások azonosítása a változások egy különleges megadását teszik lehetővé. Az *entitások rendszereként* történő megadás esetén a teljes rendszert entitásokra bontjuk fel, amelyek kapcsolatban lehetnek egymással (hathatnak egymásra), és ahol minden entitás a változását a bemenete és a saját állapota alapján határozza meg, illetve ezek alapján hathat a kimenetére.

Entitások-rendszereként a teljes rendszer változását egy adott formájú *meghatározottságként* adhatjuk meg, ahol a belső transzformációs szabályok alapján „automatikusan” történik a változás. A meghatározottságként történő megadás egy absztrakt megadású rendszer, azaz a rendszerek egész csoportjának a változását képes megfogalmazni.

Az entitás állapota egyrészt hatással van önmagára; ebben az esetben a hatás átvitele a megfelelő belső mechanizmus segítségével történik. Kapcsolódó entitások esetén bizonyos hatások átvitelét azonnalinak tekintjük, pl. a kapcsoló felkapcsolására kigyullad egy izzó, a mérleg egyik oldalának lenyomására a másik oldal felemelkedik... Bizonyos kapcsolatok esetén azonban megjelenhet egyfajta közvetettség, mégpedig olyan módon, hogy a ható és/vagy a függő entitásnak kell kezdeményezni a hatás átvitelét. A hatás adott közvetettségen keresztül történő átvitele esetén így megkülönböztethetünk *aktív* és *passzív entitásokat*. A hatás átvitelét az aktív entitás kezdeményezi, míg a passzív entitás csak válaszol az állapotára vonatkozó kérdésekre, illetve „elszenvedi” az egyes transzformációkat.

Például egy „aktív” pályázó „entitás” benyújt egy pályázatot egy hivatalnak, amivel adott folyamat lezajlását váltja ki. A pályázat eredményéről (mint absztrakt állapotáról) érdeklődhet; ebben az esetben ő az aktív fél. Másrészt a pályázat eredményéről kiértékelhetik; ebben az esetben a pályázó passzív. Siker esetén a megfelelő szerződés megkötése után hozzájut az adott forráshoz (passzív).

A két időpontjára leszűkített rendszert *állapotváltásnak* neveztük; ha eltekintünk az időpontoktól, akkor *átmenetnek*. Amennyiben tényleges állapotváltozás történt, akkor az az *esemény*.

Egy adott entitás tényleges állapotváltása (ami egy belső esemény), a külső entitások számára (külső-) eseményként jelenik meg, amely saját állapotuk megváltoztatására kényszerítheti őket. Az összekapcsolódó entitásokban így egy-egy változás egész eseménysorozatokat, pontosabban esemény-rendszereket, így egyben átmeneteket válthat ki.

Entitások rendszerében a következő eszközök szükségesek az *átmenetek megadásához*:

- az entitásnak
 - válaszolnia kell az absztrakt kiterjesztett állapotára vonatkozó kérdésekre, ill.
 - le kell tudnia kérdezni a környezetében (minimálisan a bemenetén) elhelyezkedő entitások absztrakt állapotát
- az entitásnak lehetővé kell tennie az absztrakt állapotának a módosítását
 - ez történhet az állapotmódosításra történő közvetlen felkérésre
 - vagy bizonyos jelzett eseményekre való közvetett reagálásra
- az entitásnak jeleznie kell az absztrakt állapotának megváltozását a kimenete felé

Aktív entitás esetén adott *hatás felvétele* a következő elemekből épül fel:

- az entitás lekérdezi a bemenete (absztrakt) állapotát
- a bemenet és a saját absztrakt állapota alapján esetlegesen állapotot vált
- ha történt „lényegi” állapotváltás, akkor ezt közölheti („jelezheti”) a kimenetén lévő entitásokkal

Érdemes megfigyelni, hogy adott esetben elegendő a bemenetnek csak azokat az absztrakt állapotait figyelembe venni, amely az entitás aktuális absztrakt állapota alapján tényleges változást eredményezhetnek. Hasonlóan, elegendő lenne csak azokat a „lényegi állapotváltásokat” közölnie, amely a kimenetén lévő entitások együttes környezetének ismeretében azokban tényleges hatást eredményezhet.

Aktív entitás esetén adott *hatás érvényesítésének* az elemei a következők:

- vagy az entitás
 - közli az (esetlegesen megváltozott) absztrakt állapotát, mint egy külső eseményt,

- majd a függő entitás teszteli a környezetét és elvégzi az esetleges módosítást
- vagy az entitás
 - a saját állapota és a függő entitás saját és környezetének állapota alapján közvetlenül végrehajtja a függő entitás módosítását.

A hatás érvényesítésének kétfajta módjában a közvetettség (azaz a szabadság-fok), illetve a beavatkozás erősségében láthatunk különbséget. Az első módszer nagyobb önállóságot és döntési jogkört hagy a függő entitásnak, míg a második egy kifejezett állapot-mederbe terelést kényszerít ki. Az első így variábilisabb, nagyobb változatosságú rendszert eredményez, míg a második szorosabb korlátozásokat képes fenntartani (!).

Passzív entitás esetén az *entitás hatásának érvényesítési* módja a következő:

- az entitás válaszol az absztrakt állapotára vonatkozó kérdésekre

Passzív entitás esetén az entitást ért *hatás felvételének* a lehetőségei:

- közvetve: az entitás fogadhat (külső) eseményeket, melyre
 - a saját és környezete állapotának függvényében esetlegesen megváltoztatja állapotát,
 - esetleges akcióval a kimenetén további hatásokat válthat ki (közvetve egy eseménnyel vagy közvetlenül egy módosítással).
- közvetlenül:
 - az entitás lehetővé teszi, hogy kívülről megváltoztassák az állapotát
 - ennek következményeként közvetve vagy közvetlenül további hatásokat válthat ki

Az entitások rendszerét így olyan meghatározottságként tudjuk megadni, amely alapvetően adott feltételek mellett bekövetkező állapot-transzformációk rendszere. A feltételek az entitások absztrakt állapotaira vonatkoznak, a transzformációk pedig esetleges értesítéseken keresztül állapotmódosításokat

eredményeznek. A közvetettség, a nagyobb szabadságfok több (esemény-) értesítést alkalmaz, nem pedig a másik entitás állapotváltozásának a kikényszerítését.

A nagyobb szabadságfok nagyobb variabilitást tesz lehetővé.

§ Entitások egymásrahatásának megadása. Az entitásokra felbontott rendszert, azaz az egymáshoz kapcsolódó entitások rendszerét így egy hármas eszközkészlettel adhatjuk meg. Az entitások rendszerében egy-egy entitás:

- (esemény) a belső vagy külső eseményének hatására
- (belső-transzformáció) állapotot válthat a saját és környezete állapotának függvényében,
- (akció) melyről értesítheti környezetét vagy kikényszerítheti annak változását

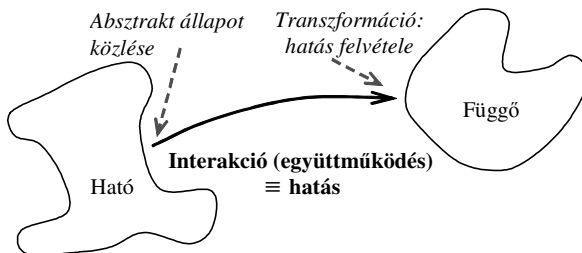
Az összekapcsolódó entitások működését a következőkben interakciónak vagy együttműködésnek nevezzük.

▽ Interakció vagy együttműködés. A rendszer részei közötti *interakciót* (más néven *együttműködést*) adott, hatónak nevezett részeiktől adott, függőnek nevezett részek felé történő információ-továbbításként valamint a függő részekeken végrehajtott állapot-transzformációként definiáljuk, mely során a ható részek közlik absztrakt kiterjesztett állapotaikat és adott (absztrakt) módon beavatkozhatnak a függő részek állapotaiba, ill. változásaikba. Az interakciók a részek közötti hatásokat reprezentálják, azaz az interakció és a hatás fogalmát ekvivalensnek tekintjük.

Az interakció a *ható részek oldaláról* tekintve az *absztrakt kiterjesztett állapotuk közlését* jelenti. Rendszer egy része önmaga és bemenetének állapotán kívül nem rendelkezik a rendszer más részeire vonatkozó információval és – természetesen – a rendszeren kívüli információval sem. A résznek ezért csak a kiterjesztett állapot információja áll rendelkezésre, ezért nem is lehetséges, hogy ennél több információt közöljön. Adott jellegű rész *absztrakt állapota* értelmezhető *absztrakt hatásként*, amelyben elvonatoztattunk a függő résztől, illetve annak állapotában okozott változástól.

Az interakció a *függő részek oldaláról* tekintve a *hatás felvételét* jelenti, mely esetén megváltozhat a részek állapota. A rész változása legfeljebb

önmagára és a kimenetére terjedhet ki. Adott jellegű rész esetén *adott jellegű hatás felvétele* egy *absztrakt hatás*, melyben eltekintünk a ható résztől, illetve annak állapotától.



29. ábra. Interakció (hatás).

Tevékenység

Az objektumorientált módszertanok az *objektum változását* a következő módon írják le. Az objektum adott időpillanatban valamely állapotban van. Az állapotban valamilyen tevékenységet végezhet, például várakozhat. Valamely (külső vagy belső) esemény bekövetkeztével, vagy automatikus átmenet esetén a tevékenység befejeződésével az objektum ugyanabba vagy egy másik állapotba léphet, mely átmenetet időtartam nélkülinek tekintjük. Az átmenethez kapcsolható „örszem”, amely adott attribútumokra megfogalmazott logikai állítás. Ebben az esetben csak az állítás teljesülése esetén hajtodik végre az átmenet.

Startpontú kvantumrendszerek esetén a változás megadható a kezdő időpont állapotából a következő időpont állapotába, majd ismételten az ezután következő időpontok állapotába váltó transzformációs lépések sorozataként. A transzformációk sorozatával szimulálhatjuk a rendszeren belüli hatások eredményeként létrejött változást.

A teljes rendszer egy transzformációját a rendszer adott entitása szempontjából vizsgálva a transzformáció a (belső-) transzformáció és az akció komponenseire bontható, melyben az átalakulást a rész önmagára kifejtett belső hatása, illetve a környezet külső vezérlése határozza meg. Az akció a környezet esetleges transzformációjának a lehetősége, azaz a külvilág entitásai számára jelent egy esetleges átalakulást.

Magát a transzformációt időtartam nélküli műveletnek tekintjük, ami az adott (kiterjesztett) állapot alapján meghatározza a vizsgált időpontban érvényes állapotot. Egy transzformáció például úgy alakítható időtartammal rendelkező műveletté, azaz *tevékenységgé*, ha a transzformáció után a rendszer „megvárja” a kvantumrendszer következő vagy egy későbbi időpillanatának a bekövetkeztét. (Transzformáció *előtti* várakozással is megadhatunk tevékenységet, ez azonban nehezebben áttekinthető modellt eredményez.) Ez alapján a következő modell készíthető: adott időpontban a rendszer egy entitása a környezet hatása, valamint az önmagára kifejtett hatás alapján megváltoztathatja az állapotát, amely változás az adott állapotból a következő ütem állapotába váltó transzformációként jelenik meg. Ugyanakkor az időpontban felvett állapot hathat az entitás környezetére, azaz annak állapotát is megváltoztathatja, amely a környezet entitásainak esetleges állapotváltásaiként, azaz transzformációjaként jelenik meg. A transzformációt, illetve az azt kiváltó hatásokat időtartam nélkülinek tekintve a változás olyan elemi tevékenységek sorozataként adható meg, amelyek a hatások okozta transzformációt követően „megvárják” a következő időpillanatot.

Az átmenet összetett transzformációját (függvényét) *elemi transzformációs* lépések sorozatává (eleminek tekintett függvényekből képzett összetett függvénné) bonthatjuk. Ha az entitás egy kompozit, azaz felírható további részletek kompozíciójaként, akkor a transzformációt egy többdimenziós függvény adja meg. Ebben az esetben lehetséges, hogy az *összetett transzformáció* egy elemi transzformációja egyetlen részletre vonatkozik. Különálló részletekre vonatkozó transzformációs elemek szimulálhatók az összetett függvények sorrendiségeként (először *a* részlet módosítása, majd *b* részlet módosítása). Azonban ebben az esetben egyik transzformáció sem használja fel a másik eredményét, ezért ezek végrehajtása tetszőleges sorrendben, elvben párhuzamosan is történhet. Azaz *összetett transzformáció* megadható külön részekre vonatkozó transzformációk független kompozíciójaként (ebben az esetben lényegtelen a sorrend), valamint transzformációk függő kompozíciójaként, mely esetén lényeges a sorrend, mivel a transzformáció egy lépése felhasználja a korábbi lépések eredményét.

Egy összetett transzformáció végrehajtása közben (például egy transzformáció-sorozat egy közbülső lépése esetén) az entitás, vagy az entitások együttes rendszere csak részlegesen megváltozott, azaz *inkonzisztens állapotban* lehet. Az entitás állapota a transzformáció végrehajtása közben nem

értelmezett, így azt fel sem lehet használni a szükséges transzformációk meghatározásához, ezért az entitást ill. entitásokat le kell *zárni* az összetett transzformációk végrehajtásának idejére.

▼ **Tevékenység.** Egy időtartam nélkülinek tekintett, ezért félbeszakíthatatlan transzformáció és az azt követő, következő ütemig tartó várakozás együttesét *tevékenységnek* nevezzük. A transzformáció lehet összetett: ebben az esetben az altranszformációk függő vagy független kompozíciójaként áll elő. A transzformáció lehet üres (identikus): ekkor a tevékenység egyszerűen a következő ütemig történő várakozás.

Elemi tevékenységekből *összetett tevékenységet* képezhetünk. Egy összetett tevékenység transzformációk és esetleges akciók olyan sorozataként jelenik meg, amelyek egymástól várakozással, ütemekkel vannak elválasztva. A módszertanoktól eltérően megközelítésünkben egy *elemi tevékenység nem szakítható félbe*, mivel az egy időtartam nélkülinek, így félbeszakíthatatlannak tekintett transzformációt jelent, amely eredménye a következő időpontban jelenik meg. Elemi tevékenységek sorozatából képzett összetett tevékenység (legegyszerűbb esetben egy több ütemig tartó várakozás) azonban bármely elemi tevékenység üteme után félbeszakadhat, például egy külső esemény bekövetkezésének hatására.

▼ **Időhatár.** Az entitásnak valamely szigorúan monoton (halmozódó vagy csökkenő értékének) a kizárólag az időtől függő változását és az arra vonatkozó értékkorlátok együttesét *időhatárnak* nevezzük, amennyiben az adott értékkorlát átlépése hatást eredményezhet az entításra vagy annak környezetére. *Elemi időhatár* egyetlen értékkorlátot jelent.

☞ Az *összetett (több értékkorlátot tartalmazó) időhatár megadható elemi időhatárok sorozataként.*

☞ Az *elemi időhatár egy adott időpontig történő várakozásként jelenik meg*, ami valójában a halmozódó vagy fogyó mennyiség szignifikáns megváltozásának (absztrakt értékváltozásának) a képe.

☞ *Kvantumrendszer esetén a legkisebb időhatár a következő időpontig történő várakozás*, mivel a „korábbi” időpontokban a rendszer nincs értelmezve.

§ Az időhatár lejárta egy eseménynek felel meg, ahol a bázisállapothoz képest az értékkorlát átlépése jelenti a tényleges (absztrakt) állapotváltást.

☞ *A tevékenység elkezdődése egy eseményt határoz meg*

☞ *A tevékenység befejeződése egy eseményt határoz meg*

☞ *A tevékenységek időtartamhoz kötődnek*, mivel várakozást is tartalmaznak, míg a transzformációk „azonnali” műveletek.

Összetett tevékenységként egy több lépésből álló, adott végrehajtási idejű munkafolyamatot adhatunk meg. A tevékenység végrehajtása az entitás speciális (absztrakt) állapota, így a tevékenység elhagyása (befejezése vagy félbeszakadása) is egy másik állapotnak felel meg.

§ A tevékenység egy állapot-kettőst határoz meg, a végrehajtó entitás a tevékenység megkezdésével a tevékenység végrehajtásának megfelelő absztrakt állapotba kerül, majd a tevékenységből való kilépéssel elhagyja azt az állapotot, ezért egy másik absztrakt állapotba vált. Szükség esetén a kilépés utáni állapot tovább bontható, hogy a tevékenység befejezése vagy félbeszakítása történt-e meg.

§ A tevékenység és az absztrakt állapot fogalmai egymásnak megfeleltethetők. A tevékenység egyrészt egy absztrakt állapotot jelent: a tevékenység elkezdésével az entitás az absztrakt állapotba vált, a tevékenység elhagyásával pedig az absztrakt állapotot is elhagyja, s egy másik állapotba vált át. Másrészt az entitás az absztrakt állapotba való belépést követően az állapotot egy esemény (például egy időhatár túllépése) eredményeként hagyja el, ami egy tevékenységnek felel meg.

Entitások rendszereként felírt változás esetén az absztrakt állapotoknak az a szerepe, hogy meghatározza bizonyos függő entitások változását. A tevékenység és az absztrakt állapotok kapcsolata azt jelenti, hogy bizonyos entitások változása függhet attól, hogy adott entitás éppen végrehajt egy adott tevékenységet, vagy hogy már befejeződött-e (esetleg félbeszakadt) egy-egy tevékenysége.

▼ **Szinkronizáció.** A tevékenységek befejeződésének vagy félbeszakadásának (együttesen: „végződésének”) eseményéig történő várakozást szinkronizációnak nevezzük.

Absztrakt állapot, esemény és módosító

Összekapcsolódó entitások rendszereként megadott változást hatásokként, illetve – annak speciális eseteként – tevékenységekként adunk meg. A hatás az állapot esetleges megváltozását, adott mederbe terelését kényszeríti ki, mégpedig a ható elemek állapotainak a függvényében.

Az entitások rendszerében minden entitás ön maga és környezete állapotának, azaz a kiterjesztett állapotának a függvényében változtathatja meg az állapotát. Mivel egy kiterjesztett entitás több különböző állapotkombinációja is ugyanazt a hatást eredményezheti, ezért az entitás szempontjából az azonos csoportba tartozó állapotok összevonhatók *absztrakt állapotokba*. Ha az entitást a környezetére tett hatásai alapján vizsgáljuk, akkor az azonos hatás kiváltása alapján meghatározhatjuk az entitás absztrakt külső állapotait, amely a konkrét állapotok adott csoportjai.

Entitás absztrakt állapotainak a meghatározása valójában az állapotthalmaz adott *szempont* szerinti felosztásait követik, így az absztrakt állapotok egymással bonyolult kapcsolatokat alkothatnak.

Az állapotok absztrakt állapotokba történő csoportosítása egy korábbi technikánkat, a *lépésenkénti véges választású közelítés* módszerét követi. Amennyiben startpontú determinisztikus kvantumrendszerek entitásainak a változásait tiszta átmeneti függvények segítségével írjuk le, akkor mindössze a (kiterjesztett) állapotból a következő időpont célállapotát meghatározó állapot-transzformációt kell megadnunk. Azaz, mindössze a kiterjesztett állapotra kell elvégeznünk az osztályozást. A lépésenkénti véges választású közelítés módszerének egy egyszerűbb változatát alkalmazhatjuk az absztrakt állapotok meghatározására.

▼ **Részrendszer állapotának lépésenkénti véges felosztása.** A rendszer adott része, pl. egy környezetével együtt vett „kiterjesztett” entitás állapotára végezzük el a következőt: A ténylegesen felvett állapotokat csoportosítsuk véges számú diszjunkt részalmazokba, osztályokba. Amennyiben még kettő vagy több osztályt kapunk, a következő lépésben bizonyos osztályok összevonásával újabb osztályozást határozzunk meg. Legfeljebb véges

számú lépésben végezhetjük el a csoportosításokat, mivel az első lépésben véges számú felosztást engedélyeztünk, a további lépésekben pedig csökkennie kell az osztályok számának. A részrendszer állapotait csoportosító osztályok-alosztályok rendszerét a részrendszer állapotának lépésenkénti véges felosztásának nevezzük.

☞ *Az állapotok lépésenkénti véges felosztása esetén minden állapot-osztály, illetve egyelemű osztályokként tekintve minden kiindulási állapot egy-egy absztrakt állapotot határoz meg.*

§ **Az állapotok lépésenkénti véges felosztása egy szempont,** amennyiben az állapotokat adott időpont szituációinak tekintjük. A szempont esetén az $i0$ a konkrét állapotokat határozza meg, utolsó $i\infty$ lépésként pedig a tényleges állapotok teljes halmazát kapjuk.

§ **A kizárólag diszjunkt vagy tartalmazás viszonyban álló véges számú absztrakt állapotok legalább egy lépésenkénti véges felosztást határoznak meg.** Ekkor a diszjunkciós és tartalmazási viszonyok alapján fel lehet építeni (esetleg több változatban) a lépésenkénti véges felosztás hierarchiáját, mely kiindulási szintjén a ténylegesen felvett állapotok, mint egyelemű osztályok helyezkednek el, az utolsó szinten a teljes halmaz, a közbülső szinten pedig a diszjunkt állapotok, melyeket esetleg ki kell egészíteni az adott szintre vonatkozó „egyébként” absztrakt állapottal, amely a teljes halmaz és a diszjunkt halmazok uniójának a különbsége.

☞ *Egyetlen, nem a teljes tényleges állapothalmazt megadó absztrakt állapot egy lépésenkénti véges felosztást határoz meg, ahol az egyetlen köztes szinten az s absztrakt állapotnak megfelelő csoport, illetve a teljes halmaz és az absztrakt állapotba tartozó állapotok különbségének $\neg s$ csoportja, mint két osztály szerepel.*

☞ *Egy s nem tetszőleges („valódi”) absztrakt vagy konkrét állapot közvetve egy $\neg s$ „valódi” absztrakt állapotot is meghatároz.*

Mivel az állapotok több szempont szerinti felosztását is elkészíthetjük, ezért az absztrakt állapotok között bonyolult viszonyok állhatnak fenn:

☞ *Az entitás bizonyos absztrakt állapotai más (kevésbé absztraktabb, azaz pontosabb) absztrakt állapotok gyűjtőfogalmai, szituációkként tekintve a következményei lehetnek. Ezért a rendszer adott része esetén egyszerre több, következmény viszonyban álló absztrakt állapot is teljesülhet.*

☞ *Mivel több szempont is meghatározható, ezért a rendszer adott részére egyszerre több (nem feltétel vagy következmény viszonyban lévő) absztrakt állapot is teljesülhet, mely absztrakt állapotok különböző felosztások eredményei.*

☞ *Minden absztrakt állapothoz tartozó, a felosztás előző lépésében kapott osztályok kizáróak, így ha a (csoportosító) absztrakt állapot teljesül, akkor a felbontás egy absztrakt alállapotának teljesülése a többi osztály nem-teljesülését vonja maga után. Ez a kizárásos elv közvetlenül is igaz a lépésenkénti véges felbontás legelső lépéseként kapott osztályozásra.*

☞ *Egy csoportosító absztrakt állapot teljesülése esetén, ha az pontosan két alosztályt csoportosít, akkor az adott alosztályhoz tartozó absztrakt állapot teljesülése a másik alállapot nem-teljesülését vonja maga után, és fordítva, azaz ebben az esetben az egyik alállapot felhasználható a másik definiálására.*

Az absztrakt állapotok szoros kapcsolatban vannak a logikai értékű függvényekkel, mivel azok egy-egy kiválasztást határoznak meg.

§ Minden absztrakt állapot kölcsönösen megfeleltethető a tényleges állapotok halmazán értelmezett adott, logikai értékű függvénynek. Az absztrakt állapothoz rendelt függvény igaz értéket ad minden, az absztrakt állapot csoportjába tartozó állapot esetén, egyébként pedig hamisat. Ellentétes irányban: a tényleges állapotok halmazán értelmezett logikai függvény egy absztrakt állapotot határoz meg, mivel bármely függvény az alaphalmazon egy osztályozást indukál; logikai függvény esetén azonos csoportba kerülnek azok az állapotok, amelyekhez igaz érték, s a másik csoportba kerülnek, amelyekhez hamis érték rendelődik.

☞ *A tényleges állapotokon értelmezett logikai értékű függvény alkalmas egy absztrakt állapot definíciójára.*

☞ *Minden, a tényleges állapotokon értelmezett logikai értékű függvény egy egyszerű szempontot határoz meg, amennyiben az állapotokat adott időpont szituációinak tekintjük.*

☞ *Az absztrakt állapot értelmezhető logikai értékű függvényként, amely minden tényleges állapot esetén meghatározza, hogy ez az állapot az absztrakt állapothoz tartozik-e.*

§ Absztrakt állapotokból a logikai műveletek segítségével összetett absztrakt állapotokat képezhetünk.

Egy kiterjesztett entitás ténylegesen felvett állapotainak halmaza alapján az absztrakt állapotok több, különböző rendszerét is megadhatjuk, azonban mindössze azok a „lényegiek”, amelyek befolyásolhatják az entitás, illetve a kapcsolódó entitások működését. Tehát az absztrakt állapotokat a kapcsolódási módok vizsgálata alapján célszerű meghatározni, majd azokat az egymáshoz való viszonyaik szerint osztályozási hierarchiákba csoportosíthatjuk.

A környezete szempontjából az entitásnak azok az absztrakt állapotai a „lényegiek”, amelyek hatással vannak a környezet entitásaira. Passzív entitás esetén a hatás az állapotra vonatkozó kérdés megválaszolásaként jelenik meg, aktív entitás esetén pedig az állapotváltozásával kapcsolatos eseményként vagy a környezetének közvetlen módosításaként. A környezet által is ismert állapotokat általában *publikus absztrakt állapotoknak* nevezzük.

Az entitás ugyanakkor bizonyos belső műveletsorozatot, esetleg párhuzamosan több műveletsorozatot is végezhet. Az entitás kívülről is észlelhető változása például leírható a megfelelő belső műveletsorozattal, azonban csak bizonyos esetekben és bizonyos pontokon felel meg kívülről is észlelhető állapotváltozás. A belső műveletsorozat megadásához is szükség lehet bizonyos absztrakt állapotok azonosítására, például egy adott állapotváltásnak feltétele lehet bizonyos, pl. két tevékenység befejeződését jelző absztrakt állapotok elérése (szinkronizálás). Ezek a *belső absztrakt állapotok* csak közvetve jelennek meg külső állapotként, de alkalmasak az entitás belső változásának absztrakt leírására.

Az állapot tényleges megváltozását eseménynek neveztük. Az absztrakt állapot tényleges megváltozása így az absztrakt esemény, amely több konkrét esemény csoportosítását jelenti.

A környezetéből vizsgálva egy adott entitás megváltozása *külső eseményként* jelenik meg, míg az entitás oldaláról ez a saját változás egy belső esemény.

A környezet szempontjából ezért elegendő az entitásnak csak azon absztrakt állapot-változásait megadni külső eseményként, amelyek „lényegiek”, azaz a környezetben változást idézhetnek elő, melyeket *publikus eseményeknek* nevezünk. A *belső események* nem feltétlenül eredményeznek publikus eseményt, esetleg csak valamely belső eseménysorozat vagy műkö-

dés eredményeként, azonban alkalmasak az entitás belső folyamatainak leírására, azaz az entitás változásának egy absztrakt, elvi szintű megadására.

A hatás a függő entitás állapotának esetleges módosulását jelenti, a hatás felvétele így egy állapot-transzformációval adható meg, amely a függő entitás (saját és környezetének) kiterjesztett (absztrakt) állapotának vizsgálata alapján végrehajtja az állapotmódosítást.

Az entitások rendszerében az entitások összekapcsolása történhet közvetlen módon: ekkor az állapotát megváltoztató entitás a tőle függő entitásokon közvetlenül hajtja végre az esetleges módosításokat. Ez a közvetlen kapcsolódás, amely egy kisebb szabadságfokú, merevebb rendszert eredményez.

A nagyobb variabilitást megengedő közvetett mód alapvetően az „értesítések” rendszerét jelenti: a megváltozott entitás eseményként közli állapotának („lényegi”) megváltozását; az értesítéseket „figyelő” entitások ennek hatására elvégezhetik a szükséges állapotmódosításokat. A közvetett módnak lehetővé kell tenni a ható-függő entitások összekapcsolását, „összeszövését”.

Az értesítés alapú rendszereknél az összekapcsolás első eleme az adott entitás külső eseménye, amely így aktív módon, az absztrakt állapot változásának azonnali következményeként eredményezhet hatást. Az ilyen jellegű absztrakt állapotokat ezért *publikált absztrakt állapotnak* nevezzük, amelyen azt értjük, hogy az állapotba lépést figyelhetik más entitások, így arról azok azonnali értesítést kaphatnak.

Mivel a tevékenységek szoros kapcsolatban vannak az absztrakt állapotokkal, ezért a *publikált tevékenység* fogalma is értelmezhető, azaz a tevékenység elkezdéséről és befejeződéséről, avagy félbeszakadásáról értesítést kaphatnak az azt figyelő entitások.

A külső vagy belső események hatására történő állapotváltozásokat *módosítókként*, más néven *állapot-transzformációkként* adhatjuk meg. A legyszerűbb módosító közvetlenül váltja ki egy adott absztrakt állapotba történő állapotváltást, általában azonban a környezet és a saját absztrakt állapotának függvényében határozza meg a felveendő állapotot. A tiszta átmeneti függvények formájában felírt változás esetén valójában egyetlen, a kiterjesztett állapoton értelmezett összetett állapot-transzformáció adja meg a változást. Ezt az összetett transzformációt felbonthatjuk több, a környezetének és önmaga adott absztrakt állapotait tesztelő és az esetleges állapotváltozást egy-egy absztrakt állapot szintjéig meghatározó módosítókra.

Mivel az állapot-transzformációt a környezet és az entitás állapota határozza meg, ezért az *összetett állapot-transzformáció felbontását* alapvetően a következő módokon végezhetjük el:

- a környezet, illetve az entitás részei alapján, így külön megadhatók az adott résztől függő, illetve az adott részre vonatkozó transzformációk
- az entitás absztrakt állapota alapján: az entitás az absztrakt állapotától függően (a környezete és a saját további állapotainak vizsgálatával) más és más absztrakt állapotba válthat; az állapota alapján az entitás megváltoztathatja a viselkedését, például más-képpen reagálhat ugyanarra az eseményre
- az entitás belső eseményei, azaz állapotváltozásai alapján; ez az állapotváltás közvetlen transzformációval is megvalósítható, azonban a független megadás egy áttekinthetőbb modellt eredményezhet
- a környezet eseményei alapján: a környezet (absztrakt) állapotának megváltozása az entításra, illetve közvetve annak környezetére is hatással lehet, ami módosítóként, illetve akcióként jelenik meg.

A részleges állapot-transzformációk *absztrakt állapot-transzformációk*, mivel mind csak adott absztrakciós szintig írják le az entitás reakcióját, így azokat – a további megfelelő részleges állapot-transzformációkkal – ki kell egészíteni, amíg a szükséges teljes állapotváltozás meghatározható.

Egyes esetekben az absztrakt állapot-transzformáció megadása teljes mértékben is leírhatja a szükséges változtatást, pl., ha egy esemény csak korlátozott mértékű reakciót vált ki, esetleg ha a reakciót kevésbé befolyásolja az entitás állapota.

Az egyszerű vagy különböző összetettségű absztrakt állapotmódosítók a hatás felvételének absztrakt módjait határozzák meg. Így alkalmasak annak a leírására, hogy az entitás milyen jellegű hatások eredményeként változtathatja meg az állapotát, azaz tekinthetők az adott eseményeket „figyelő” módosítóknak, amelyek az (absztrakt) esemény hatására a kiterjesztett állapot vizsgálata alapján hajtja végre az átmenetet.

A tiszta átmeneti függvénnyel megadott rendszer esetén az *entitást* a *környezete szempontjából* a következő módon írhatjuk le:

- publikus absztrakt állapotai, amelyek a környezet entitásaira passzív módon is hathatnak (az absztrakt állapotok lekérdezhetők).
- publikus absztrakt eseményei, pl. publikált (más elemek felé közölt) absztrakt állapotváltásai, amelyek aktív módon hathatnak a környezetre
- a kívülről beállítható absztrakt állapotok; a környezet ható entitása a módosító segítségével közvetlenül kiválthatja az entitás állapotának megváltozását
- „figyelő” absztrakt állapot-transzformációk: az entitás ezeknek a segítségével felveheti a többnyire eseményekként megjelenő hatásokat

Az *entitás belső működése* további elemek megadását igényli:

- belső absztrakt állapotok
- belső absztrakt állapot-transzformációk
- tevékenységek
- belső absztrakt események (pl. belső állapotváltások jelzése, időhatárok túllépése vagy tevékenység befejezése...)

Az entitás kívülről észlelhető absztrakt állapotai, ill. eseményei a hatásokat a ható oldaláról írják le, az absztrakt transzformációk (pl. „figyelők”) pedig a függő elem oldaláról.

Az entitások közvetett összekapcsolása így történhet az absztrakt esemény hatására történő absztrakt transzformációk meghatározásával.

Vezérlés: működés algoritmikus környezetben

Valós világ objektumai esetén az állapotok változása és az objektumok egymásra hatása *automatikusan*, a világ saját törvényei alapján meghatározott módon zajlanak le. Ha a levegő hőmérséklete fagyponthoz alacsonyodik, befagy a tó vize, a viharos erejű szél ágakat törhet le, stb.

Ha az objektumokat egy algoritmikus környezetben (egy Turing-jellegű gépen, például számítógépen) reprezentáljuk, akkor a törvények automatizmusát szimulálni kell az algoritmikus környezet által megkövetelt módon. Az algoritmusok jellemzője, hogy valamely cél elérésének módját adják meg, azaz bizonyos irányultságon, *szándékosságon* alapszanak. A törvények automatizmusát ezért algoritmikus környezetben szándékossággal kell szimulálnunk. Példánkban a levegőnek közölni kell hőmérsékletét a tóval, amely adott érték alatt „befagyott” állapotba vált át. Másik megoldásként a levegő az adott érték alá csökkent hőmérséklet esetén utasítja a tavat a befagyásra. Lehetséges azonban az irányultság megfordítása is: a tó is lekérdezheti a levegő hőmérsékletét és „dönthet” a befagyásáról. Az első és harmadik változat a valósághoz közelebbi modellt ad, a második viszont hatékonyabban működik, mivel közelebb áll az algoritmikus környezet sajátosságaihoz. A szimuláció úgy is megvalósítható, ha egy algoritmus ismételt módon teszteli az állapotokat és a törvények alapján végrehajtja a szükséges módosításokat. Ez a reprezentáció áll a legközelebb a valósághoz, de a legkevésbé hatékony. Ebben az esetben az algoritmus szándékossága a törvények, vagy más szóval: a korlátozók betartásával fogalmazható meg. Erre a szemléletre épülnek a korlátozókon alapuló (*constraint based*) programozási nyelvek.

A szándékosság eszköze a kibernetika egyik legalapvetőbb fogalma, a *vezérlés*, amely eredménye a vezérelt dologra kifejtett hatás. A vezérlés egy asszimetrikus kapcsolatot jelent: a vezérlő *irányítását* a vezérelt dolog felett.

Algoritmikus környezetben adott hatást kétfajta módon reprezentálhatunk. Hatást vagy a vezérlő fejt ki a vezérelt objektumra, vagy az objektum lekérdezi a rá ható objektum (absztrakt) állapotát és végrehajtja a szükséges állapotváltást.

A változás algoritmikus reprezentációja esetén ezért a működő objektumnak folytonosan tesztelni kell önmaga állapotát, illetve környezetének (bemenetének) állapotait, hogy hatnia kell-e környezetére (kimenetére), illetve környezete (bemenete) kényszeríti-e állapotának megváltozására. A folytonos tesztelés egyszerűsíthető, ha csak az állapot változása esetén ellenőrizzük, hogy a változás kihat-e más objektumok állapotára, így a változás eredményezi („hajtja végre”) a hatást.

Rendszer algoritmikus környezetben történő reprezentációja esetén a hatások törvényekben megfogalmazható automatizmusát szándékossággal,

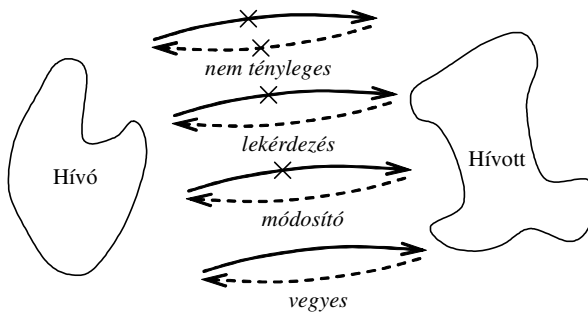
azaz vezérléssel kell szimulálnunk. Ebben a környezetben ezért alapvető jelentőségű, hogy a rendszer egy része rendelkezik-e a vezérléssel, vezérel-e egy másik részt, illetve egy másik rész vezéreli-e a vizsgált részt.

Algoritmikus reprezentáció esetén a rendszer részei közötti, törvényekben megfogalmazható hatások automatizmusát szándékosan, vezérléssel kell szimulálnunk. A következőkben a hagyományosnak tekinthető függvényhívás/visszatérés vezérlési szerkezettel megvalósítható interakciókat vizsgáljuk. Ebben az esetben a hívó a vezérlő, amely a vezérlést átadja a hívott programrésznek, amely a vezérelt. A hívó és a hívott részt is működőnek tekintjük, mivel a hívó nem szakította félbe, hanem csak felfüggesztette működését.

A hívás-visszatérés vezérlési szerkezet esetén a hívott adott műveletét (alprogramját: azaz függvényét vagy eljárását) tekintve a vezérlő és a vezérelt objektum között a következő interakciós lehetőségek léteznek:

- A művelet nem ad vissza értéket és mellékhatást sem fejt ki (a várakozást is mellékhatásnak tekintjük). Ez az üres művelet valójában egy üres (identikus) transzformáció, amely nem tekinthető tényleges interakciónak.
- A művelet nem fejt ki mellékhatást, de visszaad valamely értéket. Az objektum csak önmaga és környezete (bemenete) állapotát ismeri, így ez az interakció-típus az absztrakt kiterjesztett állapot *lekérdezését* jelenti. Ez egy egyirányú, a hívottól a hívó felé történő információ-továbbítás.
- A művelet nem ad vissza értéket, de mellékhatást fejt ki. Az ilyen típusú művelet a *transzformáció* vagy *módosító*. Ez egy egyirányú, a hívótól a hívott felé történő információ-továbbítás.
- A művelet mellékhatást fejt ki és értéket is utal vissza. Ez a változat a módosító és a lekérdezés kombinációjaként is tekinthető, amely egy kétirányú információ-továbbítás, egy információcsere. Mivel a visszautalás egyetlen, nem ismétlődő érték meghatározását jelenti, ezért ez a *vegyes* változat megadható egy kezdő módosító, egy lekérdezés és egy záró módosító elem-hármasával, melyek közül a lekérdezés határozza meg a visszaadott

értéket, valamint a kezdő vagy a záró módosító közül az egyik elhagyható.



30. ábra. Hívás-visszatérés interakciók lehetséges típusai.

VI

Belső rendszer

A rendszert a következőkben entitások együttes működéseként vizsgáljuk. A szempontok absztrakciói segítségével megállapítható a rendszerünk elemei közötti hasonlóság, illetve kohéziós erő. Bizonyos elemek együttesen reagálhatnak adott hatásokra, ami utalhat a hasonlóságon alapuló implicit, vagy a hatásokon alapuló explicit kapcsolatra. A hatások általánosításaként kapjuk a szabályokat, melyek segítségével a rendszert meghatározottságként adhatjuk meg.

A belső rendszer

Szemléletünkben a rendszer a változás teljességét jelöli; a rendszeren kívül nincs semmi más, nincs további külvilág vagy „külső rendszer”, amellyel a rendszerünk kapcsolatban lenne. Egy részekre bontható rendszer, azaz egy kompozit esetén külön választhatjuk a rendszer bizonyos részeit, melyeket együttesen a rendszer *entitásának* nevezünk, s ezzel kapcsolatban értelmeztük a „környezet” és a „külvilág” fogalmakat. Eddigi vizsgálatainkban a rendszer egyetlen entitását emeltük ki és annak viszonylatában elemeztük a rendszer változását, hogy mely részek hathatnak az entitásra, a hatást az entitás milyen módon veheti fel, valamint, hogy az entitás milyen módon hathat a környezetére. Az absztrakt állapotok, absztrakt események, a transformációk és a tevékenységek négyes eszközkészletével megadhatjuk az entitás változását, illetve, hogy annak során az hogyan hat a környezetére, illetve hogyan reagál annak változásaira.

Egyetlen entitás változásának, viselkedésének ilyen módon megadott képe kiterjeszthető úgy, hogy a teljes rendszer változását egymásra ható entitások meghatározottságaként adjuk meg, azaz a teljes rendszer entitásokba csoportosított részeinek együttes működésű belső rendszereként. A *belső rendszer* vizsgálatához az entitások egymáshoz való *viszonyát* kell meghatároznunk.

Struktúra

Egy (absztrakt megadású) rendszeren végrehajtott absztrakció eredményeként módosulhatnak a rendszeren belüli hatások. Esetenként hamis hatások („babonák”) is megjelenhetnek. Azonban, ha az absztrakción nem bővíti az elemek változásának tartományait, azaz ha *tartomány-absztrakciót* alkalmazunk, akkor legfeljebb gyengülhetnek, esetenként el is tűnhetnek a hatások, azaz ekkor bővíthetnek az invarianciák.

Egy tartomány-absztrakció segítségével meghatározhatunk, „körvonalazhatunk” egy rendszer egy adott helyéhez tartozó entitást. Az absztrakció eredményeként kapott absztrakt rendszerben a helyhez még mindig kapcsolódó további részek együttese jelenti az így meghatározott entitást.

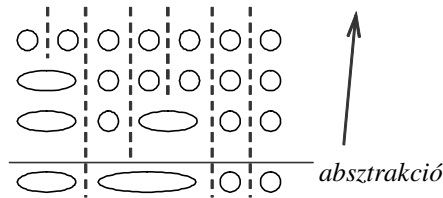
A részek közötti kapcsolatok szempontjából tehát különleges szerepűek a tartomány-absztrakciók. A megfelelő szabályokat teljesítő tartomány-absztrakciókat szempontba rendezhetjük, mely szempont minden közbülső absztrakciója egy tartomány-absztrakció.

✓ **Tartomány szempont vagy strukturálási szempont.** Az $i\infty$ kivételével tartományabsztrakciókat tartalmazó szempontot *tartomány szempontnak* vagy *strukturálási szempontnak* nevezzük.

A hatások tranzitívak, azaz ha egy A rész hat B részre, mely B pedig C részre hat, akkor közvetett módon A is hat C részre. Egy kompozit rendszer részeit a hatások alapján egymáshoz kapcsolódó diszjunkt „szigetekbe” csoportosíthatjuk, ahol az egymáshoz nem kapcsolódó részek különböző „szigeteken” helyezkednek el. Bizonyos rendszerek esetén az összes rész egyetlen szigetre kerül, azaz minden rész kapcsolatban van az összes többi résszel. Ha egy tartomány-absztrakciót hajtunk végre, akkor az növelheti az invarianciákat, amely úgy jelenik meg, hogy a szigeteken belül újabb zárt csoportokat tudunk azonosítani, mely csoportok elemei (ezen az absztrakciós szinten) függetlenek más csoportok elemeitől. Egy újabb tartomány-absztrakció végrehajtásával a csoporton belüli alcsoportokat határozhatunk meg, és így tovább...

Érdemes megjegyezni, hogy mind a „szigetek”, mind azok csoportjai és alcsoportjai a rendszer entitásaként jelennek meg.

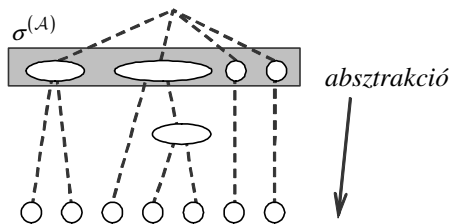
§ Strukturálási szempont a rendszer hierarchikus felbontását állítja elő. Egy (absztrakt megadású) rendszer esetén az egymáshoz kapcsolódó részek a részek diszjunkt halmazait határozzák meg. Egy strukturálási szempont bármely köztes (tartomány) absztrakciója bármely korábbi absztrakt képhez tartozó résznek olyan diszjunkt részhalmazait állítja elő, melyek esetén az egymáshoz már nem kapcsolódó részek külön részhalmazba kerülnek. Azaz a strukturálási szempont a rendszer hierarchikus felbontását állítja elő, ahol az egy szinten lévő, egymáshoz nem kapcsolódó részek diszjunktak, a nagyobb értékhez tartozó „felsőbb szintű” kép esetén pedig egy rész további független alrészekké bontódhat szét.



31. ábra: Strukturálási szempont alkalmazásával a rendszer független entitásokra bomlik.

▽ Dekompozíció. Egy tartományabsztrakció eredményeként kapott részek diszjunkt halmazait az eredeti rendszerre visszavetítve a rendszer *tartomány-absztrakció szerinti dekompozíciójának* nevezzük.

▽ Szempont szerinti struktúra. Egy rendszernek egy strukturálási szempont szerint az invarianciák alapján történő felbontása eredményeként kapott hierarchikus entitás-alentítások rendszerét a rendszer adott *szempont szerinti struktúrájának* nevezzük.



32. ábra: Struktúra, mint hierarchikus dekompozíció.

Egy strukturálási szempont absztrakcióit követve a rendszer egymástól független részeire hullik szét, illetve egy rész további független alrészeire bontható.

▼ **Szempont szerinti strukturálható rendszer.** A *szempont szerinti strukturálható a rendszer*, ha a strukturálási szempont alkalmazása esetén megjelenik új invariancia, azaz a szempont valamely köztes absztrakciók szinten egy korábbi részt kettő vagy több alrészeire bont fel.

Tipizálás, fogalmi rendszer

Egy strukturálási szempont alkalmazása esetén a rendszer egymástól független részekre hullik szét. Ugyanakkor, egy (tetszőleges, nem csak strukturális) szempont mind magasabb absztrakciói során az eredeti rendszer hasonló, de kezdetben még különböző részletei közül egyre több válik egymással ekvivalenssé, egyre több sorolható azonos csoportba. A csoportok száma egyre csökken, miközben a rendszer mind nagyobb részleteit foglalják magukba. Az egyre átfogóbb típusok szintén egy hierarchikus rendet alkotnak, mégpedig az absztrakción sorozat végrehajtásának ellenkező sorrendjét követve. A hierarchia legalján a kiindulási rendszer részei helyezkednek el.

Az elemek azonos csoportba sorolását úgy végezzük el, hogy két absztrakt részrendszert azonos csoportba tartozónak tekintünk, ha az egyik megfelelő pozícionálásával a másik rendszerrel azonosnak tekinthető rendszert kapunk.

Vegyük sorra az ezzel kapcsolatos korábbi definícióinkat! A hasonlítható rendszerek közötti átjelölést *pozícionálásnak* nevezzük, ha az mindössze a rendszerek időpontjait és helyeit jelöli át. A rendszer (esetleg identikus) korlátozásának következménye a *rendszer tényleges fogalma*. Rendszer tényleges fogalmának közvetlen pozícionálása a *rendszer lehetséges fogalma*. Általában véve pedig a rendszer *fogalmának* neveztük azt a rendszert, amely egy tényleges fogalom pozícionálásával azonosnak tekinthető.

A fogalom koncepciója, illetve a pozícionálás technikája lehetővé teszi, hogy egy rögzített rendszer részeinek azonos jellegét megfogalmazhassuk (ebben a fejezetben általános, azaz nem csak absztrakt megadású rendszereket vizsgálunk):

✓ **Fogalmi azonosság.** Két (hasonlítható) rendszert *fogalmilag azonosnak* tekintünk, ha ez egyik megfelelő pozícionálásával a másikkal azonosnak tekintett rendszert kapunk. Jelölés σ_A és σ_B rendszerek esetén: $\sigma_A \widetilde{\Leftrightarrow} \sigma_B$.

Határozzunk meg egy kompozitban diszjunkt részeket! Például határozzuk meg egy tartomány absztrakciót és vegyük a rendszer arra vonatkozó dekompozícióját. Egy strukturálási szempont alkalmazásával a mind absztraktabbá váló részek közül egyre több válik fogalmilag azonosná.

✓ **Fogalmi következmény, absztraktabb fogalom.** Ha egy σ_A fogalomnak létezik olyan pozícionálása, amely σ_B feltétele, akkor azt mondjuk, hogy a σ_B a σ_A *fogalmi következménye* és a $\sigma_A \widetilde{\Rightarrow} \sigma_B$ jelölést alkalmazzuk. A σ_B *absztraktabb fogalom* a σ_A fogalomnál, ha fogalmi következménye, de fogalmilag nem azonosak.

Egy szempont rögzítése esetén definiálhattuk a hasonlítható rendszerek egymástól való eltérését, sőt, kiterjesztett valós szempont esetén ez egy távolságot határoz meg. A fogalmi azonosság segítségével az eltérés, illetve a pontosság definíciói a fogalmakra is kiterjeszthetők.

✓ **Fogalmi eltérés, fogalmi pontosság vagy hasonlóság.** Egy rendszer $\sigma_{j \in J}$ fogalmainak adott $\prec$ szempont szerinti *eltérése* (jelölés: $\widetilde{\Delta}_{j \in J}^{(\prec)} \sigma_j$) az a legkisebb olyan $d \in D_{\prec}$ érték, melyre minden $j \in J$ esetén a $\prec_{(d)}(\sigma_j)$ rendszerek fogalmilag azonosnak tekinthetők. A $\sigma_{j \in J}$ fogalmak $\prec$ szempont szerinti *pontossága*, vagy más néven azok hasonlósága (jelölés: $\widetilde{|\sigma_{j \in J}|}_{\prec}$) a $\prec_{(d)}(\sigma_j)$ fogalom.

☞ *Rendszer fogalmainak fogalmi pontossága (más néven fogalmi hasonlósága) is a rendszer fogalma.*

☞ *Rendszer fogalmainak fogalmi pontossága a vizsgált fogalmak fogalmi következménye.*

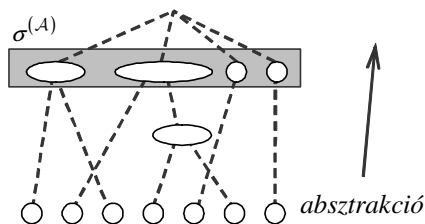
✓ **Hasonlóbb fogalmak.** Adott $\prec$ szempont, valamint egy rendszer σ_A , σ_B és σ_C fogalmi (például egy kompozit részei) esetén a σ_A és σ_B

hasonlóbb a σ_C fogalomnál, ha a σ_A és σ_C , valamint σ_B és σ_C hasonlósága egyaránt absztraktabb fogalom σ_A és σ_B hasonlóságánál.

Amennyiben a rendszer részeit fogalmakként tekintjük, a részek egyre absztraktabb képei egymással mind hasonlóbbakká válnak, egyre több lesz közöttük olyan, hogy az egyik absztrakt kép egy pozicionálással valamely másik absztrakt képnek megfeleltethető. A szempont így egy absztrakciós hierarchiát határoz meg, ahol adott absztrakciós szinten azonos csoportba sorolhatjuk a fogalmilag azonossá vált részeket. A nagyobb absztrakciós szinten legfeljebb bővülnek ezek a csoportok, ami olyan absztraktabb, egyben bővebb csoportként (pontosabban fogalmi következményként) jelenik meg, amely bizonyos szempontból tartalmazza a kevésbé absztrakt csoportokat. A szempont szerinti, fogalmilag azonossá váló részek meghatározását fogalmi tipizálásnak vagy röviden tipizálásnak nevezzük.

✓ (Fogalmi) tipizálás. Adott $\prec$ szempont esetén egy rendszer diszjunkt részeit tekintve (fogalmi) *tipizálásnak* nevezzük a részek egyes absztrakciós szinthez tartozó képeinek csoportosítását a fogalmi azonosság alapján.

☞ *A tipizálás eredménye egy hierarchikus csoportosítás.*



33. ábra: (Fogalmi) tipizálás.

A tipizálás érdekessége, hogy a strukturáláshoz hasonlóan ugyancsak a részek hierarchikus csoportosítását határozza meg. Strukturálás esetén azonban a magasabb absztrakciók esetén invariáns részeire hullik szét a rendszer, míg tipizálás során a magasabb absztrakciók egyre bővebb csoportokat, egyre absztraktabb fogalmakat határoznak meg.

A szempont egyre erősebb absztrakciónak alkalmazása során az eredeti teljes rendszer egyszerre hullik szét részeire, miközben az egyes absztraktabbá váló részek egyre átfogóbb fogalmakba csoportosíthatók. A művelet a rendszert egyszerre bontja fel és gyűjti egybe. A folyamat során a teljesség egyszerre válik partikulárisrá és absztraktabbá, ezzel magyarázható az absztrakció és az aggregáció „közelség” szempontjából vett dualitása.

▼ **Fogalmi rendszer.** Hasonlítható $\sigma_{i \in I}$ rendszereken (azaz $\Sigma^{(\sigma_{i \in I})^*}$) adott $\prec_{j \in J}$ szempontok absztrakcióinak alkalmazása esetén vegyük a kapott eredményrendszereket, melyeket a fogalmi azonosság erejéig egy-egy ekvivalencia-osztályba tartózóaknak tekintünk. A kapott fogalmak halmazát a $\Sigma^{(\sigma_{i \in I})^*}$ rendszerek $\prec_{j \in J}$ szempontok szerinti fogalmi rendszerének nevezzük: (jelölés: $\Sigma_{\prec_{j \in J}}^{(\sigma_{i \in I})^\sim}$).

☞ *Mivel a rendszer fogalmi hasonlíthatók, így a hasonlítható rendszerek fogalmi is hasonlíthatók, ezért a fogalmi rendszer fogalmi is hasonlíthatók.*

☞ *Egy fogalmi rendszer valamely eleme nem feltétlenül minden rendszer fogalma, mivel fogalmat a rendszer tényleges fogalmából képzünk (pozicionálással).*

Mivel a fogalmak, illetve azok pozicionálása, azaz alkalmazása is hasonlítható rendszereket eredményez, ezért azokon alkalmazhatók például a konjunkció és a diszjunkció műveletei. Azaz a rendszer bizonyos részeit (amelyek szintén fogalmak) leírhatjuk elemibb fogalmak segítségével.

Hatás ellensúlyozása, kohézió

Egy strukturálási szempont alkalmazása során a rendszer független, invariáns részeire hullik szét. Bizonyos részek hamarabb válhatnak el egymástól, míg a részek adott csoportjai még magasabb absztrakciók esetén is megtartják kapcsolataikat. A szempont így alkalmas a részek összekapcsolódási erejének, azaz a kohéziós erőnek a megadására.

▼ **Kohéziós erő.** Egy (absztrakt megadású) rendszer $\sigma_{i \in I}$ részei közötti adott $\prec$ strukturálási szempont szerinti *kohéziós erő* az a legkisebb

olyan $d \in D_{\prec}$ érték, melyre minden $i \in I$ esetén a $\prec_{(d)}(\sigma_i)$ rendszerek invariánsak.

☞ *Független részek közötti kohéziós erő minimális (i0).*

☞ *Adott strukturálási szempont esetén a nagyobb kohéziójú részek később válnak szét.*

A definíció alapján így már beszélhetünk egy rendszer adott szempont szerinti szorosabban vagy gyengébben kapcsolódó részeiről.

Vegyük egy (absztrakt megadású) rendszer adott entitását, amelyre hat egy másik entitás! A hatás a vizsgált függő entitás állapotváltozását adott „mederbe” terelheti, azaz a függő entitás viselkedése megváltozhat. Átmene-tekként megadott rendszer esetén a hatás eredményeként (amelyet a ható entitás (absztrakt) állapota vált ki) a függő entitás megváltoztathatja az álla-potát. A függő entitás eredeti és módosult állapota közötti eltérés egy meg-felelő absztrakció végrehajtása során „eltűnhet”, azonossá válhat. Azaz az absztrakció elnyelheti, „ellensúlyozhatja” az entitást ért hatást. Másik hatás esetén esetleg gyengébb vagy erősebb absztrakció szükséges a hatás „ellen-súlyozásához”. Azaz a végrehajtandó absztrakció erőssége alkalmas az entitást ért hatás mérésére.

▽ **Hatás erőssége.** Egy (absztrakt megadású) rendszerben az E enti-tást ért *hatás* $\prec$ szempont szerinti *erőssége* az a legkisebb olyan $d \in D_{\prec}$ érték, melyre $\prec_{(d)}(E) \Leftrightarrow \prec_{(d)}(E')$, ahol E' az entitásnak a hatás eredmé-nyeként módosult képe. Ennek megfelelően beszélhetünk adott szempont szerinti erősebb, illetve gyengébb hatásról.

☞ *A hatás erőssége az eredeti és a hatás eredményeként megváltozott működések, mint hasonlítható rendszerek eltérése.*

§ **Hatások rögzített kiterjesztett valós szempont szerinti erőssége távolság** az absztrakt megadású rendszer egy rögzített entitásának különböző paraméterekhez tartozó működései esetén (az azonosnak tekinthető rendsze-rek ekvivalencia-osztályain), mivel a hatás erősségét hasonlítható rendszerek eltérésére vezettük vissza.

A hatás erősségét tehát az eredeti és a módosult változás eltéréseként értelmezzük, azaz a nagyobb eltérés nagyobb hatást jelent.

✓ **Esemény fontossága vagy lényegessége (szignifikanciája).** Egy (átmeneti függvényként megadott) rendszer E entitásának állapotváltozásai, azaz eseményei hatást eredményezhetnek az entitás környezetében. Azaz: az entitás konstans változásához tartozó működéshez képest eltérhet a rendszer működése. Az *esemény* $\prec$ szempont szerinti *fontossága* az a legkisebb olyan $d \in D_{\prec}$ érték, melyre a két működés azonosnak tekinthetővé válik. Ennek megfelelően beszélhetünk adott szempont szerint fontosabb, illetve kevésbé fontos eseményről.

✓ **Külső esemény fontossága.** Egy entitás szempontjából vett külső esemény adott szempont szerint fontosabb, ha a szempont szerint nagyobb eltérést okoz az entitás változásában.

Az esemény, illetve külső esemény fontossága esetünkben tehát nem annak gyakoriságával vagy ritkaságával van összefüggésben, hanem, hogy az mekkora hatást eredményez.

Ahogy a definíciók is mutatják, a hatás erőssége és a kohézió fogalmi közös alapkoncepcióra épülnek. Ezt úgy is elképzelhetjük, hogy egy entitást ért bizonyos hatások megszüntethetnek az entitás valamely részei közötti kapcsolatot – mégpedig (egy rögzített szempont szerint) a szorosabban kapcsolódó részek leválasztásához erősebb hatás szükséges.

Például tekintsünk egy szobát, amelyben két szék közvetlenül egymás mellett, egymást érintve helyezkedik el. Ha megfelelő irányba megtoljuk az egyik széket (hatás), akkor a két szék együtt mozdulhat el. A szoba többi része független, míg a két szék között van valamekkora kohéziós erő. Ha ellentétes irányba toljuk el a vizsgált széket, akkor az elmozdulása független lesz a másik széktől, a hatás tehát megszüntette ezt a kohéziós erőt. A „szobában van két egymás melletti szék” helyett a „szobában van két szék” absztraktabb kép ellensúlyozni tudja ezt a hatást. Ha egy nagy erőt fejtünk ki a székre, akkor az széttörhet. A törés valószínűleg az illesztések mentén történik, így letörhet a szék lába vagy a karfája. A hatást még nagyobb absztrakcióval kellene ellensúlyoznunk, pl. „a szobában két szék alkatrészei vannak”. A szék lábának egyes részei szorosabban kapcsolódnak egymáshoz, mint a szék többi részéhez, azaz azon belül nagyobb a kohéziós erő, sokkal

erősebb hatás szükséges a részeinek szétválasztásához. A szék egyes fémrészei, fémlapok vagy a csavarok esetén még erősebb a kohézió. Igen nagy erő szükséges például egy fémlapban elhelyezett csavar, sőt, még nagyobb egy csavar fejének az eltávolításához. Ez a „dekompozíciós” tevékenység-sorozat (amin most a székek szétverését értjük) a rendszer adott szempont szerinti struktúráját adja meg: *székek(szék(karfa, háttámla, ülőke, láb1, láb2, láb3, láb4), ...)*. Az egyes részek természetesen tovább bonthatók. A struktúra egyben az adott szempont szerinti gyengébb-szorosabb kohéziót is megadja. Egy másik szempont szerinti tipizálás azonos csoportba vonhatja a székeket, illetve azok közös alkotórészeit, vagy azonos csoportba sorolhatja az (azonos vagy különböző székekhez tartozó) széklábakat.

Implicit és explicit kapcsolat

Rendszer két entitása közötti kapcsolat az egyik entitás hatását jelenti a másik entításra és/vagy a másik hatását az elsőre. A hatás a függő entitás működésének (változásának) megváltozásaként jelenik meg, mégpedig feltételes („ha-akkor”) módon: azaz *ha* a ható entitás adott módon működik, *akkor* az a függő entitás működését adott mederbe tereli.

Átmeneti függvénnyel megadott változás esetén korábban láttuk, hogy egy kiválasztott entitás szempontjából a hatás a következőképpen jelenik meg:

- a *környezet* entitásainak *állapota* (lekérdezés) vagy annak megváltozása (*esemény*) esetén
- az entitás (absztrakt) *állapotának* figyelembe vételével mindez
- az entitás *transzformációját* (azaz módosulását) okozhatja,
- amely *akciót* (külső entítások megváltoztatását) eredményezhet.

A teljesség kedvéért jegyezzük meg, hogy a változás eredményeként az entitáson belül egy változás sorozat, azaz egy *tevékenység* is megkezdődhet, illetve módosulhat, amely tevékenység későbbi akciókat is kiválthat.

Átmeneti függvénnyel megadott változás esetén így a hatás az állapotok, pontosabban az absztrakt állapotok alapján fejt ki az eredményét.

A hatás bizonyos együttes változásként jelenik meg a külső szemlélő számára, azaz a ható entitás változására a függő is megváltozik.

Ha a kompozit rendszer entitásához hasonló entitás a rendszer valamely más helyén is megjelenik, azaz a két entitás fogalmilag azonos, akkor a közös környezetük adott hatására mindkét entitás azonos módon reagálhat. A külső szemlélő számára mindez úgy jelenik meg, hogy az entitások szinkronban változnak, egyszerre módosulnak. Látszatra az entitások kapcsolatban vannak egymással, holott a valóságban lehetséges, hogy azok között nincs is tényleges kapcsolat. A ható „külső entitás” szempontjából a két entitás között létezik valamiféle kapcsolat, hiszen az általa kifejtett hatásra azonos módon reagálnak. Ugyanez mondható el, ha a két entitás azonos módon hat valamely „külsőként” tekintett entitásra.

Vizsgáljuk meg, hogy átmeneti függvénnyel megadott változás esetén adott absztrakciók alkalmazását követően hogyan módosulnak az entitások közötti hatások!

A hatást az entitás állapota, illetve annak megváltozása, azaz az entitás eseménye okozza. Mivel bizonyos állapotok, illetve események ugyanazt a hatást fejthetik ki, ezért az állapotok és események adott hatások szempontjából csoportokba, azaz absztrakt állapotokba, illetve absztrakt eseményekbe csoportosíthatók. A hatás eredménye pedig a függő entitás adott állapotát meghatározó kényszerként, átmeneti függvény esetén az állapot transzformációjaként jelenik meg. A hatás tehát már önmagában is meghatároz egy absztrakciót, mivel a ható entitás állapotait csoportokba sorolja, a függő entitásnak pedig egy adott absztrakciós szintig meghatározza az eredmény-állapotát. Így kimondhatjuk a következő megállapításokat:

§ Kapcsolat, mint a rendszer következménye. Adott entitások közötti adott kapcsolatok a rendszer következményét, azaz a rendszer egy absztrakt képét határozza meg, amely az entitások állapotait a hatások és azok eredmény-korlátozóinak pontosságáig adja meg, valamint eltekint a rendszernek a kapcsolódó entitásokon kívüli részeitől.

§ A hatás a rendszerre vonatkozó információ, mivel hatás esetén a korlátozó nem lehet a tetszőlegesség.

Tekintsük most az entitások egy kiválasztott csoportját, valamint egy külső entitást, amely hathat az entitásokra, illetve függhet azoktól. A külső

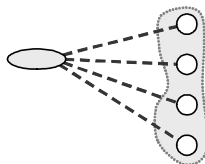
entitás bizonyos absztrakt állapota/eseménye azonos jellegű változást eredményezhet az entitásokon: mindegyik entitásra korlátozóként fogalmi szinten azonos absztrakt állapotot határozhat meg. A megfelelő absztrakció végrehajtása esetén a különbségek akár el is tűnhetnek, így a külső entitás adott hatásait tekintve mindegyik ugyanúgy reagál, mindegyik ugyanolyan-nak tűnik.

A fentiek analóg módon alkalmazhatók a külső entitásra ható entitások esetén is.

§ Hasonló entitások azonos módon kapcsolódnak más entitásokhoz. A (fogalmi szinten) hasonló entitások azonos jellegű módon kapcsolódnak egymáshoz vagy más „külső” entitásokhoz, amennyiben a fogalmi hasonlóság szempontjának alkalmazása nem módosítja a hatást.

§ Egy entitáshoz azonos módon (ható és/vagy függő) kapcsolódó más entitások fogalmi hasonlóságot határoznak meg, ahol a hasonlóság szempontját a kapcsolatok által meghatározott absztrakció adja meg.

✓ **Explicit és implicit kapcsolat.** Entitások fogalmi hasonlósága esetén azok azonos módon kapcsolódhatnak más entitásokhoz, azonos jellegű hatást fejthetnek ki vagy azonos módon reagálhatnak azok hatásaira, amelyek így a külső szemlélő számára kapcsolatban lévőnek tűnnek, melyet *implicit kapcsolatnak* nevezünk. Az entitások közötti tényleges kapcsolatot megkülönböztetésül *explicit kapcsolatnak* is nevezzük.



34. ábra: Implicit kapcsolat, mint fogalmi hasonlóság.

Vegyük most egy kiválasztott entitást, valamint keressünk vele kapcsolatban lévő további entitásokat! A hatást a ható entitás állapota határozza meg és az a függő entitás esetleges állapotmódosulásaként jelenik meg. A vizsgált entitás hatása eltérő lehet az egyes entitásokra, azonban egy absztrakció alkalmazása eltüntethet bizonyos eltéréseket. Az így fogalmilag azo-

nossá váló entitások (amelyek eredetileg ezért fogalmilag hasonlóak voltak) ezen az absztrakciós szinten már azonos módon reagálnak a hatásra. És fordítva: ha a kiválasztott entitásra valamelyest különböző módon hatnak más entitások, azaz azok állapotai valamennyire különböznek, akkor egy absztrakció ugyancsak eltüntetheti a különbséget. Az így kapott hatás (tartományabsztrakció esetén, ill. amennyiben az még változatlanul létezik) azonban már csak gyengébb korlátozót fogalmazhat meg a függő entitásra.

Az absztrakció így elmosza a különböző hatások bizonyos egyedi jellegét, így a ható entitások absztrakt állapotainak különbségét, illetve a függő entitások pontos állapotváltozásait. Az absztrakció során így egyre több azonos jellegű hatást figyelhetünk meg a rendszerben.

Cseréljük most ki a rendszer egy entitását valamely másik entitásra! Fogalmilag azonos entitások esetén a különbség észre sem vehető. Fogalmilag különböző entitások esetén a működés megváltozása egy megfelelő absztrakcióval ellensúlyozható, mely absztrakciónak az entitás kapcsolatainak megváltozását kell eltüntetnie, így az entitás más jellegű állapotváltozásait, illetve azok más jellegű hatásait. Az eredeti és a helyettesítő entitáshoz nem kapcsolódó részek, azaz azok külvilága esetén nem módosul a rendszer működése. Most az entitást cseréljük ki egy újabbra, majd egy másikra. A rendszer működése más és más módon térhet el, amely különböző absztrakciókkal ellensúlyozható.

Egy szempont absztrakciót tekintve, amennyiben egy adott entitás helyére más és más entitásokat helyettesítünk, úgy azt kapjuk, hogy a rendszer működése a szempont szerint kevésbé fog eltérni az eredetitől, ha a szempont szerint fogalmilag hasonlóbb entitást helyettesítünk be.

§ Kompozit entitását hasonlóbb entitással helyettesítve a rendszer változása kevésbé fog eltérni az eredeti változástól a fogalmi azonosság szempontja szerint.

Mindez a helyettesíthetőség elvéhez vezet el:

§ Helyettesíthetőség elve: előre rögzített absztrakció esetén egy entitás másik entitással való helyettesítéskor a működések absztrakt képei azonosnak tekinthetők maradnak, ha a két entitás absztrakciója fogalmilag azonos, azaz a két entitás fogalmi hasonlósága megfelel az absztrakciós szintnek.

Együttműködés

Átmeneti függvénnyel megadott, entitásokra bontott rendszer esetén egyetlen entitás bizonyos eseményei hatást fejthetnek ki valamely más entitásra, amely hatás közvetve további és további entitásokra terjedhet át. A ható entitás szemszögéből mindez úgy tűnik, hogy az entitások együttesen reagálnak az eseményre. Ez az együttes reakció más, mint a fogalmi hasonlóság (azaz implicit kapcsolat), amikor a hasonló entitások azonos hatásra hasonlóan reagálnak, esetleg akár szinkron módon. Ebben az esetben a kapcsolat explicit, az ténylegesen fennáll az entitások között, és pontosan ez az összekapcsolódás eredményezi az együttes reakciót. Másik különbség, hogy ekkor az entitások nem feltétlenül hasonlóan reagálnak; a reakciók különbözőek lehetnek attól függően, hogy az entitások milyen szerepet töltenek be a csoportban. Ugyanakkor elképzelhető, hogy az egyes entitások között hasonlóság, azaz implicit kapcsolat is fennáll, így azok akár azonos szerepben is megjelenhetnek. A tényleges (explicit) kapcsolat megléte azonban feltétlenül szükséges.

A külső entitás szempontjából az összekapcsolódó, összefüggő entitások együttesen, egyetlen egységként, egy *makroentitásként* jelennek meg. A makroentitás a rendszer egy entitása (egyben fogalma), amely adott entitásokból épül fel. A rendszerben a makroentitás által elfoglalt helyek az entitásai által együttesen elfoglalt helyek.

Az együttes reakció összehangolt működést jelent, mely összehangoltság valójában a makroentitáson belüli kapcsolatokat jelenti. Ilyen makroentitás esetén, mivel léteznek belső kapcsolatok, ezért létezik bizonyos belső kohézió. És fordítva: a belső kapcsolatok erőssége határozza meg a makroentitás stabilitását, hogy az mennyire képes ellenállni a külső hatásoknak.

A belső kapcsolatokkal rendelkező makroentitás természetesen több, mint a részek egyszerű összege, és ezt a többletet pontosan a kapcsolódási módok adják, illetve ezek teszik lehetővé a környezet felé történő szervezett akciót és reakciót. Az együttműködő entitások megoszthatnak egymás között nagy külső terheléseket (hatásokat), illetve együttesen nagyobb hatások kifejtésére képesek.

✓ **Szervezet és együttműködés.** Azon entitásokat, amelyek a belső kapcsolataik révén összehangoltan működnek együttesen *szervezetnek*, összehangolt működésüket pedig *együttműködésnek* nevezzük. Az összehan-

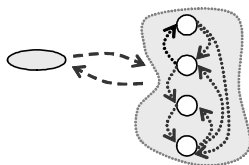
golt működés azt jelenti, hogy az entitások a belső kapcsolatrendszer alapján közvetlen vagy közvetett módon együttesen fejtenek ki hatást a külvilág felé, illetve együttesen reagálnak bizonyos külső hatásokra.

☞ *A szervezet egy makroentitás, a megfelelő környezettel, illetve külvilággal.*

☞ *Az együttes akciók/reakciók miatt a szervezet olyan egyetlen és egységes entitásként jelenik meg a környezete felé, amely a környezet entitásaira adott módon hat, illetve azok hatásaira adott módon reagál.*

☞ *A szervezetnek nem lehet része olyan entitás, amely minden további entitással invariáns.*

§ **A szervezet részei kapcsolódnak egymáshoz.**



35. ábra: Szervezet és együttműködés.

✓ **Bezárás.** (Makro)entitások bezárásának, illetve fogalmak bezárásának nevezzük azok olyan absztrakt képeit, amely egy csoportba sorolja a környezet felé azonos módon kapcsolódó entitásokat, illetve fogalmakat és azokhoz egyetlen absztrakt képet rendel.

§ **A bezárással eltekintünk az entitás pontos belső felépítésétől,** azt „fekete doboznak” tekintjük.

§ **A bezárás az entitás absztrakt modellje,** mivel az a megfelelő függvényekkel megadható. (A függvények egy csoportba foglalják a különböző, de a környezet felé ugyanúgy megjelenő működéseket, valamint eltekintenek a változásnak a környezet által nem észrevehető részleteitől.)

§ **A bezárás az entitás korlátozója, valamint az entításra vonatkozó információ nem invariáns entitás esetén.**

§ **Helyettesíthetőség és bezárás.** Ha az entitás lecserélődik, helyettesítődik egy olyan entitással, amely bezárása megegyezik az eredeti entitás bezárásával, akkor a módosított rendszerben az entitáson kívüli részek (azaz a környezet és a külvilág) működése azonos marad.

A szervezet külső vagy belső események hatására bonyolult belső tevékenységsorozatokat végezhet el, mely tevékenységeket a belső entitások tevékenységei, valamint a közöttük lévő összetett kapcsolatrendszer valósít meg. A kapcsolatok biztosítják az összehangoltságot, azaz a szervezethez tartozó tevékenységei és akciói, valamint a külső eseményekre történő reakciói egységesen jelennek meg.

A makroentitáson belül az egyes entitások különböző helyeket foglalhatnak el, és az alapján különböző módon vehetnek részt az egyes tevékenységekben. Amennyiben bizonyos résztvevők helyére megfelelő, fogalmilag hasonló entitást helyettesítünk, akkor a változtatás ellenére a szervezet belső működése ugyanaz marad, így a környezete felé is ugyanazt a külső képet mutatja (mindkét rendszer bezárása azonos). A makroentitás logikailag ugyanazt a belső működést és külső kapcsolatokat mutathatja, ha a belső entitásai ugyan máshol helyezkednek el, azonban ugyanolyan tevékenységet hajtanak végre. A makroentitás elemei szempontjából így sokkal inkább fontos egy bizonyos „logikai elhelyezkedés”, amelyet szerepnek nevezünk.

▼ **Szerep.** Egy makroentitáson belüli entitás bezárását *szerepnek* nevezzük.

☞ *A szerep az entitás absztrakt modellje.*

☞ *A szerep az entitás korlátozója valamint az arra vonatkozó információ nem invariáns entitás esetén.*

§ **Szerepre vonatkozó helyettesíthetőség.** Egy makroentitáson belül, ha egy entitást lecserélünk egy ugyanolyan szerepű entitásra, akkor a makroentitás bezárása nem változik.

(Érdemes egy rövid megjegyzést tenni a szervezettel, illetve a stabilitással és a szabadsági fokkal kapcsolatban. A szervezet egységesen képes fellépni a környezete felé, illetve egységesen reagál annak hatásaira. A szervezet ezért adott struktúra fenntartását igényli, amely a résztvevők oldaláról adott szerepek felvállalását jelenti. A szerepek ugyanakkor korlátozók, amelyek csökkentik mind a szerepet felvállaló entitás, mind a szervezet variabilitását, szabadsági fokát. A szervezet stabilitása tehát mind a szervezet, mind résztvevőinek a berögzödéséhez, kötött struktúrájához vezethet, amely eredményeként pl. a szervezet nem lesz képes valamely új hatásra a megfelelő módon reagálni.)

Egy szervezet egységesen reagálhat valamely külső hatásra, illetve egységesen hajthat végre valamely tevékenységet vagy akciót. Az egységes működés bizonyos szinkronitást jelent, amely a szervezet belső kapcsolatainak a segítségével valósul meg. Hasonló szinkronitást figyelhettünk meg a fogalmilag hasonló, azaz implicit kapcsolatban lévő entitások esetén is. Szervezet esetén azonban a kapcsolat explicit, annak ténylegesen fenn kell állnia.

A két kapcsolódási mód azonban nem zárja ki egymást, így természetesen egy szervezetben lehetnek olyan hasonló entitások, amelyek kapcsolatban is lehetnek egymással.

Egy makroentitásban több entitás is betöltheti ugyanazt a szerepet.

▼ **Kollekció vagy gyűjtemény.** Egy makroentitás azonos szerepű entitásait *kollekciónak* nevezzük.

☞ *A kollekció egy makroentitás.*

☞ *Kollekció egy eleme felcserélhető annak egy másik elemével, a kollekció szerepének, a makroentitás belső működésének és bezárásának változatlansága mellett.*

Kollekciók esetén tükröződik az explicit és implicit kapcsolatok kettősége. Mivel a kollekció elemei azonos szerepűek, ezért bizonyos fogalmi hasonlósággal is rendelkeznek. Ezért úgy is tekinthetők, mint amelyek együttesen töltenek be egy szerepet, de úgy is, mintha a szerep többszöröződné meg.

Egy szervezetnek valamely része természetesen lehet egy alszervezet, és így tovább... Azt, hogy egy kollekció (amely maga is egy makroentitás) egyben egy (al)szervezet is, azt az dönti el, hogy az egyes részek között léteznek-e kapcsolatok.

▼ **Makroentitás absztrakt struktúrája.** Egy makroentitásnak készítsük el azt az absztrakt képét, ahol az (adott dekompozíció szerinti) entitásokat azok bezárására helyettesítjük. A kapott absztrakt képet a *makroentitás absztrakt struktúrájának* nevezzük

☞ *A makroentitás absztrakt struktúrája a makroentitás absztrakt modellje.*

§ Makroentitás megadható relációként. Vegyük a makroentitás absztrakt struktúráját, amely a szerepek szintjéig meghatározza a makroentitás működését. A makroentitás felírható egy $\mathfrak{R}(\rho_i : E_{i_j})$ elem n -essel, ahol az elem n -es $\mathfrak{R}$ neve a makroentitás bezárása, a ρ_i a makroentitás szerepkörei, az $\rho_i : E_{i_j}$ pedig az adott szerepet betöltő entitást adja meg. Az $\mathfrak{R}(\rho_i)$ reláció, ahol a makroentitás bezárása, a szerepkörök és az entitások is mind hasonlíthatók, amely reláció az adott szerepe(ke)t betöltő entitások összes kombinációi közül kiválasztja a lehetségeseket, mégpedig azokat, amely esetén az E_{i_j} entitásra teljesül a ρ_i korlátozó.

Ha egy szerepkört egy további makroentitás tölt be, akkor az elem n -esek hierarchiáját kapjuk.

Környezet

A ható entitás megváltozása a függő entitás módosulását eredményezheti. A hatásnak valamilyen úton el kell jutnia (aktív vagy passzív módon) a függő entitásig. A hatás terjedése lehet közvetlen, vagy valamely közvetítő tranzitivitásán keresztül történő közvetett hatás.

Közvetlen hatás esetén a két elemnek közvetlen kapcsolatban kell lenni. Eddigi eszközkészletünkhöz (absztrakt állapot/esemény, tevékenység és akció) fel kell vennünk a kapcsolat meghatározásának eszközeit. A kapcsolatok lehetnek időlegesek is, így transzformációként lehetővé kell tennünk a kapcsolat kialakításának és lebontásának a műveleteit is. A kapcsolódás, illetve lekapcsolódás kihathat a cél-entitás élettartamára, ami az entitás elkészítését, azaz konstrukcióját, illetve lebontását, azaz destrukcióját is jelentheti. A vizsgált rész kapcsolódhat egy kollekcióhoz mint makroentitáshoz is, így a kollekció műveleteit is lehetővé kell tennünk, azaz egy entitásnak a kollekcióhoz adását, onnan eltávolítását, valamint elemeinek együttes kezelését, pl. azok sorrávetelét, szűrését... Ezeket a műveleteket együttesen strukturális vagy infrastrukturális műveleteknek nevezzük.

§ A kapcsolódó entitások együttesen egy makroentitást határoznak meg.

§ A kapcsolódó entitások makroentitása esetén az egyes entitásoknak a kapcsolódásra vonatkozó bezárása az adott entitás szerepe.

§ Kapcsolódó entitások esetén a kapcsolat a függő entitásokra korlátozóként, illetve az entitásra vonatkozó információként jelenik meg.

☞ Az entitások közötti adott kapcsolat megadható elem n -essel, ahol az elem n -es neve a kapcsolatra utal, a szerepek pedig az entitások szerepkörei.

§ Adott jellegű kapcsolat ábrázolható relációként, ahol eltekintünk az elem n -es adott szerepeit éppen betöltő entitásoktól.

Közvetlenül kapcsolódó entitások esetén az entitás eléri a hozzá kapcsolódó entitásokat, így azokon végrehajthat transzformációkat, illetve lekérheti más entitások absztrakt állapotait, így az állapotmódosításait azok hatása alapján hajthatja végre.

Közvetve kapcsolódó entitások esetén a közvetítő kisebb jelentőségű, mint a ható és a fogadó fél. Ha a közvetítő entitások szerepétől teljesen eltekinthetünk, akkor ez a kapcsolat olyan, mintha közvetlen kapcsolódás létezne a lényegi entitások között. Például egy falat megvilágító fényforrás esetén általában lényegtelennek tekinthető minden egyes „foton-entitás”, azoknak elegendő csak az együttes viselkedését számításba vennünk.

Közvetve kapcsolódás esetén azonban gyakori, hogy a közvetítő elemek módosíthatnak a hatáson. Ebben az esetben tehát meg kell határoznunk egy közvetítő *közeget*, amely egyrészt felelős a hatások átviteléért, másrészt be is avatkozhat a hatásmechanizmusokba.

Közvetve vagy akár közvetlenül kapcsolódó entitások hatásainak leírására a következőkben ismertetésre kerül egy egyszerű, mégis szemléletes módszer, amely alapja a „környezet” mint makroentitás modellezése.

A kapcsolódó entitások valójában egymás környezetében helyezkednek el. Minden egyes kapcsolat maga is egy makroentitás, valamint a kapcsolatok egy halmaza is – egy bővebb – makroentitást határoz meg.

▼ **Környezet.** Adott makroentitást, amely adott korlátozásoknak megfelelő entitásokat tartalmazhat, *környezetnek* nevezzük.

A környezet nem egyszerűen az entitások együttese, hanem önmagában is entitás jellegű, így adott tulajdonságokkal, illetve állapotokkal rendelkezhet, reagálhat bizonyos eseményekre, lehetnek transzformációi, tevékenységei, generálhat eseményeket...

Mindezek mellett a környezet

- az „adott környezetben” elhelyezkedő entitásokat tartalmazza.
- felelős a nem közvetlen hatások átviteléért, azaz *közvetítő közeg* jellegű.
- módosíthatja az egyes hatásokat, beavatkozhat a hatás átvitelébe.

Képzeljünk el egy játékot, amely egy labirintusban játszódik! A labirintus szobákból áll, amelyek között mozoghatnak a játékosok. A szobákban különböző eszközök is lehetnek. Például, ha egy bomba van a szobában, akkor a játékos megsérülhet.

A programozás-technika, pontosabban az objektumorientált technológia klasszikus kérdése, hogy a bomba robbanását hogyan valósítsuk meg. Egyik lehetőségként a bomba robbanásának kell beindítania a játékos „sérülés felvétele” műveletét. Másik lehetőségként a játékos folyamatosan lekérdezi a bomba állapotát, s annak robbanása esetén elvégzi önmagán a sérülés miatt szükséges módosítást. A probléma ezekkel a megoldásokkal az, hogy ilyenkor a bomba és a játékos túlságosan is szorosan van összekapcsolva, amely a játék módosítása vagy bővítése esetén már rendkívüli nehézségeket okozhat.

A flexibilitást egy harmadik megoldás biztosítja, amely részben már kilép az objektumorientáltság szemléletköréből, s a környezetek elvére épít. Eszerint, megadjuk a bomba és a játékos környezetét, amely valójában a szoba. A bomba a szobának fogja „jelenteni” a robbanás eseményét. A környezet, azaz a szoba a felelős a hatások átviteléért, így sorra veszi az elemeket és közli velük „nagy nyomás és hőhatás történt ekkor és itt”. A hatás átvitelekor a környezet például számításba veheti a két entitás közötti távolságot, vagy egy „varázsszoba” csökkentheti vagy éppen megfordíthatja a hatást. A környezet a hatás átvitelekor értesíti a függő elemeket a hatásról, amelyek tovább módosíthatják azt, pl. egy páncél csökkentheti a sérülést.

Nézzünk egy másik példát! Vegyünk egy síkbeli „egyenes” objektumot. Azt, hogy egy pont illeszkedik-e az egyenesre, objektumorientált szemlélet esetén általában az egyenesnél adjuk meg. Ugyanitt vehetünk fel egy másik lekérdezést, hogy egy egyenest metsz-e egy másik egyenes objektum. Hol kell azonban megadnunk mondjuk egy egyenes és egy körvonal metszését (vagy érintését, stb.) tesztelő műveletet? Az egyenesnél vagy a körnél? A válasz itt is a környezet meghatározása, amely esetünkben maga a sík – itt érdemes felvennünk a különböző síkidomok kölcsönös elhelyezkedését meghatározó szabályokat.

Harmadik példánkban vegyünk egy oldalakkal rendelkező asztallapot, amelyen súrlódásmentesen egy golyó mozog és az oldalaknál irányt vált. A golyó mozgását modellező programot rendkívül egyszerű megírni. Vegyünk azonban most két golyót! Ekkor már nem csak az oldalakkal, hanem a másik golyóval való ütközést is ellenőrizni kell. Kis munkával még ezzel a teszttel is kiegészíthető a program. Hogyan tudjuk azonban a programot tetszőleges számú golyó mozgásának a követésére is alkalmassá tenni? A válasz itt is a környezet, illetve az abban érvényes szabályok definiálása, amely környezet ebben az esetben az asztallap. A környezetnek „jelentik” a golyók az elmozdulásukat és a környezet fogja értesíteni a „feleket” az egymással és/vagy az oldallal történő ütközésről.

A környezet tehát alapvetően más entitásokat fog egybe, amelyek között általában bizonyos hatások érvényesek; azaz a környezet többnyire ezen entitások együttes környezetének a része. Ha az entitások között éppen nincsenek közvetlen hatások, a környezettel azok akkor is kapcsolatban vannak, hiszen a környezet „eléri” azokat, hathat azokra, vagy lekérdezheti az állapotukat.

§ A szervezet egy környezet.

Azaz a környezet lazább fogalom a szervezetnél. Például, egymással közvetlen kapcsolatban nem álló entitások esetén a környezet egyirányban továbbíthat feléjük egy hatást.

§ Minden makroentitás egy környezet.

§ A teljes rendszer egy környezet.

A környezet tehát önmagában csak a makroentitásnak egy másik elnevezése. A lényeges az, hogy egy-egy környezet milyen szabályokat, azaz korlátozásokat határoz meg a benne elhelyezkedő elemekre.

A környezetet tehát úgy is tekinthetjük, mint adott szabályok gyűjteményét, mely szabályok az entitásokra, illetve a hatások átvitelére vonatkoznak.

§ Minden entitás tekinthető környezetként.

Érdemes megjegyeznünk, hogy egy környezet alkotóeleme is lehet egy környezet. Korábbi példánkban a szobák együttesen a labirintust mint környezetet jelentik.

Hierarchikus környezetek valójában egy kompozíciós hierarchiát jelenítenek, azaz makroentitáson belüli (makro)entitásokat, így ebben az esetben érvényes néhány tulajdonság:

§ **Propagáció elve.** A külső környezet (makroentitás) tulajdonságai és szabályai alapértelmezésként érvényesek az alkörnyezetben, amely azonban át is definiálhatja azokat. A külső környezet „propagálja”, azaz felajánlja a rá vonatkozó entitás-tulajdonságokat és szabályokat.

Például egy vállalat (mint környezet) az elnevezését, levelezési-, email- és web-címét „felajánlhatja” alapértelmezésként a telephelyeinek, azonban azok más értékeket is meghatározhatnak.

§ **Delegáció elve.** A környezet a fogadott hatásokat „delegálja”, azaz továbbíthatja valamely részei (entitásai, alkörnyezetei) felé, illetve egy összetett tevékenységet a részek tevékenységeivel és a részek egymás közötti hatásaival valósít meg.

A vállalat például egy projekt felvállalása és megvalósítása során különböző részegységeinek és entitásainak továbbítja a feladatokat.

A környezetekre és alkörnyezetekre való felbontás nem feltétlenül a fizikai elhelyezkedést követi. A környezet a kapcsolódó entitások együttes tere, „környezete”. Például egy újság előfizetői nem feltétlenül laknak ugyanabban a városban. A példa alapján ugyancsak megfigyelhetjük, hogy a külön-

bőző szempontok szerint meghatározott környezetek különböző hierarchiákkal adhatók meg.

A környezetek önmagukban a változások, a dinamika helyszíneit határozzák meg, így egy környezetbe beléphet egy entitás, majd elhagyhatja azt, ahogy egy játékos belép egy szobába, majd átmegy egy másikba. A környezet elsősorban a hatások átvitelének terét jelentik, mely kapcsolatok lehetnek viszonylagosan rövid élettartamúak. Az élettartam alapján érdemes megkülönböztetni a környezetek két típusát. A *permanens környezet* hosszabb időtartamig létezik, míg a *dinamikus környezet* csak egy összetett tevékenységsorozat időtartamáig (például két játékos közötti csata).

Strukturálási szintek

A teljes összetett rendszer leírásakor azt összefüggő részeire bonthatjuk fel, azaz strukturálhatjuk, így külön adhatjuk meg a részeket, illetve az azokra vonatkozó együttes szabályokat. A felbontást tartomány-absztrakciók segítségével végezzük el, így a rendszer dekompozícióit kapjuk.

A strukturálásnak több szintje is meghatározható.

▼ **Reguláris struktúra.** Egyetlen tartomány-absztrakció eredményeként kapott részek diszjunkt halmazait *reguláris dekompozíciónak* vagy *reguláris struktúrának* nevezzük.

A reguláris struktúra a részek egymásmellettiiségeként határozza meg a rendszert.

▼ **Hierarchikus vagy környezetfüggetlen struktúra.** A tartomány-absztrakciókat tartalmazó szempont eredményeként kapott részek-alrészek egymásba ágyazott halmazait *hierarchikus* vagy *környezetfüggetlen dekompozíciónak* vagy *struktúrának* nevezzük.

A környezetfüggetlen struktúra a részek-alrészek hierarchiájaként írja le a rendszert, amely egy faként ábrázolható, azaz olyan egy-kezdőpontú irányított gráfként ábrázolható, amely minden más csúcspontjába a kezdőpontból csak egy úton juthatunk el.

✓ **Hierarchikus tipizált vagy környezetfüggő struktúra.** A tartomány-absztrakciókat tartalmazó szempont eredményeként a részek-alrészek egymásba ágyazott halmazait kapjuk. Rendeljük a szempont minden egyes indexértékéhez valamely módszert (minden szinthez ugyanazt a módszert is rendelhetjük). A szinten lévő elemeket ugyanannak az elemnek tekintjük, ha a módszer az elemekhez fogalmilag azonos rendszereket rendel. Az így kapott struktúrát *hierarchikus tipizált* vagy *környezetfüggő struktúrának* vagy *dekompozíciónak* nevezzük.

Minden szinthez a megfelelő identikus módszert rendelve, egy strukturálási szinten azonos elemnek tekintjük a fogalmilag azonos részeket.

A környezetfüggő struktúra egy olyan irányított gráfként ábrázolható, amely egyes élei azonos elemre, azaz azonos típusú elemekre vonatkozhatnak.

A környezetfüggő struktúra így olyan egy-kezdőpontú irányított gráfként ábrázolható, amely minden más csúcspontjába több útvonalon is eljuthatunk.

Amennyiben minden szinthez a strukturálási szempont megfelelő absztrakcióját rendeljük, akkor a szempont absztrakciónak végrehajtása során az egyes elemek a részeikre hullnak szét, ugyanakkor a szempont szerint egyre hasonlóbakká is válnak.

A fenti környezetfüggő strukturálás esetén, amennyiben a szempont összes absztrakcióit végrehajtjuk, akkor utolsó lépésként az entitásokból tetszőleges absztrakt rendszereket kapunk. Azaz ekkor egy hálót kapunk, olyan irányított körútmentes gráfot, amely egyetlen kezdőponttal és egyetlen végponttal (levélelemmel) rendelkezik.

Vegyünk egy környezetfüggő dekompozíciót! A felbontás egy közbülső szintjén lévő entitás számára „lefelé”, a dekompozíció irányába tekintve, egy részhálót látunk. Ha a típusokat nem tudjuk azonosítani, akkor olyan, mintha egy fát látnánk, azaz ez megfelel egy környezetfüggetlen dekompozíciónak. „Felfelé”, azaz a makrorendszerek irányába tekintve egy bennfoglalási sorozatot, a makrorendszerek egyre bővebb körét kapjuk.

▽ **Általános struktúra.** Több strukturálás eredményeként kapott hierarchikus tipizált struktúrákat együttesen *általános struktúrának* vagy *általános dekompozíciónak* nevezzük.

Általános dekompozíció során tehát több (tartomány-absztrakciókat tartalmazó) szempont szerinti felbontást is engedélyezünk. Amennyiben egy dekompozíciós szint adott entitását/entitástípusát vizsgáljuk, akkor úgy tűnik, mintha „lefelé”, azaz a dekompozíció irányába egy fát látnánk; ha az azonosságok is megfigyelhetők, akkor ez egy háló, vagy részháló. Mivel az entitás képe több szempont szerinti felbontásban is szerepelhet, ezért „fel-felé”, a makrorendszerek irányába is egy hálót, vagy az azonosságok figyelmen kívül hagyásával egy fát „látunk”.

§ **Általános dekompozíciót egy irányított körútmentes gráffal (DAG: Directed Acyclic Graph) adhatunk meg.**

☞ *Az egyes strukturálási-szintek egyre bővülő fogalmakat jelentenek, tehát egy reguláris dekompozíció egyben környezetfüggetlen, környezetfüggő és általános dekompozíció is, és így tovább.*

Az egyes strukturálási szintek kiemelt szerepűek a rendszerek ábrázolásakor. Különös nagy jelentőségű az általános struktúrához tartozó DAG, azaz irányított körútmentes gráf. Az általános jellege ugyanis azt jelenti, hogy a rendszerek vázának ábrázolásához nem szükséges általános gráfot alkalmaznunk, hanem kihasználhatjuk az irányítottság, illetve a körútmentesség előnyeit.

Egy DAG adott szintjén elhelyezkedő entitás esetén az entitás számára „lefelé”, a komponensei felé, valamint „felfelé” a makroentitások felé is egy hierarchiát lát, környezetfüggő, esetleg környezetfüggetlen módon (ha nem képes a hasonlóságok azonosítására). „Felfelé” ez az egyre bővülő, különböző szempontok szerinti makroentitásokat jelenti. Gyakran csak az a kérdés, hogy egy entitás egy adott makroentitás része-e, ill. egy entitás beletartozik-e egy makroentitás típusába. Az ilyen kérdésekhez már a teljes hierarchia ismerete sem szükséges; a „felfelé” irányban elhelyezkedő (sorozatként vagy faként megjelenő) entitásoknak elegendő mindössze a halmazát tekinteni.

Szabályrendszer

Korábban már említettük, hogy a rendszerek (működések) adott körét, azaz egy absztrakt megadású rendszert definiálhatunk bizonyos „szabályok” segítségével, „meghatározottságként”, amely a rendszerek nagy csoportjainak a leírására alkalmas úgy, hogy egy-egy megadás viszonylag kevés „helyet” (pl. rövidebb szimbólum-sorozatot) igényel. A következőkben a „szabályok”, illetve a „meghatározottság” fogalomköréit járjuk körbe.

Hogyan adhatjuk meg a változások nagy csoportját, azaz egy absztrakt megadású rendszert? Az egyszerűbb megközelítés érdekében először tekintsük az n véges értelmezett időponttal és m véges lehetséges állapotú determinisztikus hasonlítható (véges) rendszereket. Az ábrázolás legprimitívebb módja, ha felsoroljuk minden egyes időpontban az akkor felvett állapotot, ami $n \times m$ lehetőséget jelent. Például, ha egy számítógépnek az 1Gbyte-os memóriáját egyetlen napon csak század-másodpercenként vizsgáljuk, akkor az kb. 200 milliárd számjegyet jelentene, a nap minden egyes, kb. 8,5 millió századmásodpercében. A problémát a rendszer rendkívüli változatossága okozza.

Amennyiben meg tudunk határozni bizonyos korlátozókat, azaz a rendszerre vonatkozó információt, akkor máris csökkenteni tudjuk ezt az értéket. Drasztikus méretű csökkenést érhetünk el, ha a változásban bizonyos ismétlődő mintákat tudunk azonosítani, és azokat ki tudjuk emelni. A minták segítségével a nagy változatosságú rendszereket egy kisebb változatosságú leírásból, azaz bizonyos fajta meghatározottságként állíthatjuk elő.

Hogyan adhatunk meg egy rendszert meghatározottságként? A „meghatározottság” valójában egy hozzárendelést, azaz egy függvényt jelent. A meghatározottság legegyszerűbb változata az, ha valamely argumentumhalmazhoz a megfelelő működést rendeljük, azaz ekkor egy absztrakt megadású rendszert kapunk. Mivel az absztrakt megadású rendszer átírható az átmeneti függvények formájára (ahol az argumentum a rendszer egy kitüntetett időpontbeli állapota), amely „argumentum-állapota” tekinthető szituációként (azaz hasonlítható rendszerként) is, ezért a működések megadhatók $\sigma \mapsto \sigma'$ leképezéseként, ahol a párok bal- és jobboldalai mind hasonlíthatók; a megfelelő „argumentum-rendszer” (azaz egy baloldal) kiválasztása eredményként *meghatározza* a működést.

▼ **Szabály, szabály feltétele és következménye.** A leképezések mintájára tekintsük a következő módszert: vegyünk adott hasonlítható rendszereket és adjunk meg olyan $c_i \mapsto \sigma_j$ leképezéseket, ahol a c_i *feltétel* és a σ_j *következmény* egyaránt valamely hasonlítható rendszer egy fogalmának a pozícionálása (amelyek így ugyancsak a rendszerekkel hasonlíthatók). A leképezéseket úgy értelmezzük, hogy ha a rendszerre teljesül a c_i feltétel, azaz a c_i a rendszer absztrakciója, akkor a rendszerre teljesül a σ_j következmény, tehát az is a rendszer absztrakciója lesz (azaz a rendszerre vonatkozó információ, ha a σ_j nem a tetszőleges absztrakt rendszer). A c_i és a σ_j rendszerek esetén megengedjük, hogy azokat bizonyos argumentumok alapján állítsuk elő, így az $r(a_{p \in P})$ *szabály* a p argumentumok megadásával egy $c_i \mapsto \sigma_j$ leképezést állít elő.

▼ **Tényleges és absztrakt szabály, szabályrendszer.** A szabályok megadása esetén jelöljük meg a *tényleges szabályokat*, melyeknél, ha több tényleges szabályhoz tartozó leképezés baloldala is teljesül, akkor a jobboldalnak együttesen kell teljesülni, így azok nem lehetnek ellentmondók. Valamint, tényleges szabályok esetén megköveteljük, hogy azok következménye nem lehet tetszőlegesség. A nem feltétlenül tényleges szabályt *absztrakt szabálynak* nevezzük. A (paraméterezhető) szabályok együttesét, a tényleges szabályok megjelölésével *szabályrendszernek* nevezzük.

Amennyiben a szabály jobboldala egy tetszőleges absztrakt rendszer, akkor a szabály nem korlátozza a működést (így az nem tényleges szabály), tehát ilyen szabályokkal a szabályrendszer tetszőlegesen kiegészíthető, illetve azok elhagyhatók. Az ilyen szabályokat *tetszőleges absztrakt szabálynak* nevezzük.

A szabályrendszer jobboldalai az ábrázolandó rendszerre vonatkozó korlátozók, így bizonyos baloldalak teljesülése esetén (bizonyos szintig) meghatározzák a rendszert. Elképzelhető, hogy a szabályok annyira összefüggnek és olyan pontosak, hogy a rendszer teljes változását meghatározzák. Átmeneti függvényként megadott rendszer esetén, ha bizonyos tényleges szabályok baloldalai vonatkoznak a kitüntetett időpont állapotára, akkor a szabályok adott pontosságig leírják a működést.

Vizsgáljuk meg a szabályok néhány változatát!

A legegyszerűbb szabályok baloldalán egy tetszőleges absztrakt rendszer van, amelyet rövidítve így írhatunk: $\mapsto \sigma_j$. Tényleges szabály esetén ekkor a jobboldal a működés következménye, egyben korlátozója, amelynek minden esetben fenn kell állnia.

A szabály baloldalán feltételként a rendszer korlátozója szerepel. A feltételt a konjunkció műveletével pontosíthatjuk, azaz szűkíthetjük, a diszjunkció műveletével pedig bővíthetjük. A kivonás műveletével kizárhatunk bizonyos feltételeket. Ugyanezeket a műveleteket alkalmazhatjuk a leképezés jobboldalán is, így pontosíthatjuk vagy kevésbé korlátozhatjuk az eredményrendszert.

A szabályok adott csoportja konjunkcióként mind tartalmazhat egy adott korlátozót. Ugyanez érvényes lehet a leképezések jobboldalára is. Ekkor a baloldalnak, illetve a jobboldalnak ez a része kiemelhető, s a szabályok ezen szabály alszabályaiként is megadhatók.

A szabályok adott csoportja *általánosságban* vagy a baloldalon vagy a jobboldalon konjunkcióként tartalmazhatnak egy-egy adott korlátozót. Ekkor ez a rész kiemelhető, s az így kapott szabály alszabályai esetén azt érvényesnek tekintjük (*alapértelmezésként*), ha az egyes alszabályok nem módosítják a kiemelt korlátozót.

A szabályok egy egyszerű következtetést határoznak meg: „*ha*” a baloldal teljesül, „*akkor*” a jobboldalnak is teljesülnie kell. Egy teljesülő szabály esetén azonban mind a baloldal, mind a jobboldal a rendszer egy-egy következményét határozza meg, amelyek ekkor együttesen teljesülnek, azaz a kettő konjunkciója a rendszer következménye. A teljesülő (nem tetszőleges absztrakt) szabály tehát valójában a rendszerre vonatkozó tényleges információ. Így a szabály alkalmazása valójában egy helyzet, azaz egy korlátozó megjelenésének a felismerése, amely „követelmény-”, vagy más néven feltétel-részének megjelenése már utal a következmény, illetve a teljes szabály, mint korlátozó megjelenésére.

§ Tényleges szabály a rendszerre vonatkozó információ, amennyiben a bal- és/vagy jobboldala nem tetszőleges absztrakt rendszer.

§ A szabály a rendszer fogalma, a tényleges szabály a rendszer tényleges fogalma.

A szabály alkalmazása tehát nem a hagyományos értelemben vett következtetés, azaz a szabályok egymással való kombinálásának eredményeként kapott újabb állítás. A teljesülő szabály alkalmazása valójában az adott helyen és időben „megjelenő” fogalom felismerése. A baloldal, azaz a feltétel megjelenésével a jobboldal felhasználása inkább nevezhető a szabály alkalmazásának (a „tüzelés” szót is szokták használni), amely egy sokkal egyszerűbb eljárás, mint amit a tudományos nyelvezet a „következtetés” fogalmán ért.

A tényleges szabályok a rendszer bizonyos jellegű *szabályosságát* jelentik.

$A \mapsto \sigma_j$ alakú (azaz feltétel nélküli) korlátozó szabályok esetén ez a szabályosság azt jelenti, hogy a rendszer a σ_j korlátozói értelmezett időpontjaiban a rendszer nem veszi fel az összes lehetséges állapotot, hanem csak azok bizonyos valódi részhalmazát.

A $c_i \mapsto \sigma_j$ hozzárendelést meghatározó szabályok esetén a „szabályosság” a c_i és σ_j korlátozók között jelenik meg, azaz a c_i teljesülése, más néven: a c_i fogalom megjelenése a σ_j , pontosabban a c_i és σ_j együttes megjelenését is jelenti.

A szabályosságok kiemelt jelentőséget kapnak, ha azok többször is megjelennek a rendszerben, pl. más időben vagy más helyen. Többszörös megjelenés esetén az ismétlődések kiemelhetők és a tényleges megjelenési formák a szabályok alkalmazásaival helyettesíthetők.

A kiemelés csökkentheti az ábrázolás méretét, amennyiben a kiemelt szabály és annak alkalmazásai rövidebben (pl. kevesebb szimbólummal) ábrázolhatók, mint a tényleges működések; pontosan ekkor célszerű végrehajtani a kiemeléseket.

A szabályokra történő átirás nem feltétlenül jelent méretcsökkenést. Például egy programban egyszerűbb megadni a hónapok hosszait, mint a kialakulásukat követő szabályt implementálni. Ez a szabály pl. a következő: a régi római naptárban a március volt az első hónap, s onnan kezdve váltokozva 31, majd 30 naposak voltak a hónapok; az utolsó hónap, a február a fennmaradt napokat tartalmazta. A legenda szerint ezt a sorrendet Augustus császár módosította, hogy az eredetileg rövidebb, 30 napos augusztus hónap 31 napos legyen.

A szabályok egy „*ha-akkor*” jellegű kapcsolatot írnak le, hasonlóan az entitások közötti hatásokhoz. A szabály a hatás fogalmának általánosítása; ebben az esetben a feltétel és a következmény különböző absztrakciós szinteken is elhelyezkedhet. *Ezért a rendszeren belüli hatás a rendszer tényleges szabálya.* Szabályok esetén továbbá megengedjük a parametrizálást, valamint a két oldalt bizonyos fogalmak pozicionálásaként határozzuk meg.

A rendszeren belüli hatásokat az entitások szempontjából felbontottuk a környezet és az entitás állapotainak, valamint azok megváltozásának vizsgálatára, illetve a külső vagy belső transzformációkra. Adott külső esemény hatására adott külső és belső feltételek esetén reakcióként történő (belső) transzformáció és külső akció megadása egy szabályt jelent. Az entitások szempontjából történő megközelítésnek három speciális jellegzetessége van:

- Mind a feltétel, mind a következmény oldalon külön választjuk az entitásra, illetve a külvilágra vonatkozó elemeket.
- Események esetén a feltételben szereplő állapotokat nem szükséges folyamatosan „tesztelni”, hanem a lényegi állapotváltozásokról értesítést kap az entitás. Így a feltételben külön választhatók az események, illetve a tesztek, valamint a következményben a transzformációk mellett külön megadhatók a generált események.
- A szabály mind a feltétel, mind a következmény oldalon több időpontra is vonatkozhat, míg az esemény-állapot-transzformáció-akció szabályok esetén csak az aktuális, illetve a következő mérőidőpontok tekintett időpontra.

Az utolsó ponttal kapcsolatban egy további megjegyzést tehetünk. Ez a jellegzetesség az „időbeliségre” vonatkozik, ami közismert probléma a kibernetikában. Ha a szabály feltétele a kitüntetett előtti más időpontokra is hivatkozik, akkor az megoldható (vagy a külvilághoz vagy az entitáshoz kapcsolt) emlékezzettel, azaz olyan speciális állapottal, amely tárolja a korábbi állapotnak a szabály szempontjából lényeges elemeit. Ugyanez az elv alkalmazható a későbbi időpontokban bekövetkező következmények esetén is.

A kibernetikában ugyanez a probléma fordítva jelenik meg. Ha egy vizsgált „gép” működését emlékezzettel tudjuk leírni, akkor az azt is jelent-

heti, hogy csak ilyen módon tudjuk megfigyelni bizonyos belső rejtett állapotait.

§ Szabály megadása átmenetként. Kvantált rendszerek ábrázolásának szempontjából a memória alkalmazása azt jelenti, hogy *minden szabályt megadhatunk egy vagy több átmenetként*, amelyek vonatkozhatnak a memória-szerepű állapotok módosítására vagy azok feltételként való felhasználására is.

Rendszer ábrázolása

Foglaljuk tehát össze, hogy egy absztrakt megadású, azaz több lehetséges működésű rendszer milyen alapeszközökkel adható meg!

A rendszerünket egyrészt *entitásokra* bonthatjuk fel, a rendszer változását pedig szabályok meghatározottságaként állíthatjuk elő, mely szabályok feltétel, illetve következmény részeit entitásokra vonatkoztathatjuk. Az „entitás” fogalom jelenthet több entitást magába foglaló makroentitást, illetve egy entitás környezetét, vagy több entitás együttes környezetét („kontextusát”).

Egy entitás a rendszer adott helyeit foglalja el, mely helyeket együttesen *entitás-azonosítónak* tekintünk. Az entitás-azonosítók segítségével megállapítható két entitás azonossága, illetve, hogy az egyik entitás a másikban helyezkedik-e el (annak „része”).

Az entitás lényegi jellemzőit *absztrakt állapotainak* a segítségével adhatjuk meg. Az absztrakt állapot vonatkozhat az entitás valamely részére, vagy a teljes entításra.

Bizonyos állapotok meghatározására alkalmazhatjuk a „lépésenkénti véges választású közelítés” módszerét, amely egyben ábrázolja is az állapotot, a választásokhoz rendelt index-sorozatként, azaz az entitástól már elszakított *értékként*.

Az *elemi értékek*, azaz a komponenseket nem tartalmazó indexsorozatok valójában a következő csoportba sorolhatók: mennyiségek, minőség-jelzők és kódolt értékek.

- A *mennyiségek* alapvetően szám-jellegű értékek (a választások ekvidisztáns beosztása, stb.), amelyek adott, a választásra jellemző pontossággal rendelkeznek, és amelyekkel aritmetikai műveletek és összehasonlítások végezhetők. (Ugyancsak mennyiségek az idővel kapcsolatos értékek.) A mennyiségek valamely „darabra”, más néven „egységre” vonatkozó számot jelentenek, azaz a mennyiség értelmezéséhez szükséges a mértékegység is.
- A *minőségek* nem számszerűsített beosztásokat jelentenek, amelyekhez rendelt indexeket pl. egy-egy megnevezéssel jelezhetünk. A minőségeknek is van pontossága, így beszélhetünk pontosabb vagy kevésbé pontos megadásról. Pl. a „kék” és a „világoskék” esetén az utóbbi egy pontosabb megfogalmazást jelent. A logikai értékek valójában adott minőségek teljesülésére vagy nem teljesülésére utalnak.
- A *kódolt értékeket* például szabadszöveges módon adhatjuk meg, amely értelmezése, dekódolása után juthatunk el a tényleges értékhez. Például egy lakcím szöveges megadása esetén annak egyes részei az utcára, házszámra, stb. vonatkoznak. Gyakran a dekódolást az ábrázolás külső értelmezőjére hagyjuk.

Elemi értékekből összetett értékeket képezhetünk, amely kompozit értéket jelent. Az értékelemeket a szerepkörük alapján azonosítjuk – ekkor az összetett érték az értékkomponensek elem n -esével ábrázolható –, vagy azonos szerepkör esetén minden érték egyenrangú – ekkor elegendő az értékkomponenseket felsorolnunk.

A már meghatározott állapotokból logikai és esetleg aritmetikai összehasonlításokkal és műveletekkel újabb állapotokat képezhetünk. Gyakran „állapotnak” csak a logikai értékű absztrakt állapotot nevezzük, amely így meghatározza, hogy az entitásra érvényes-e az így megadott tulajdonság; ekkor a többi állapotot „tulajdonságnak” nevezzük.

Az entitás jellemzéséhez tartozik, hogy milyen további entitásokhoz kapcsolódik. A kapcsolatok valójában a hatások átvitelének az útvonalai.

A kapcsolatok a tulajdonságokhoz hasonlítanak, azonban a kapcsolatok létrejöhetnek, lebomolhatnak, amely a kapcsolódó entitás létrejöttével, illetve lebomlásával is egybeeshet. A kapcsolatok lehetnek kötelezőek vagy opcionálisak, illetve lehetnek legfeljebb egy vagy több számosságúak.

A kapcsolat nem érték-jellegű, hanem adott, egyediséggel rendelkező entitások között létezik.

Az entitás adott időpillanatban az absztrakt állapotai segítségével jellemezhető. Az entitás állapotának a megváltozása az *esemény*, adott jellegű állapotváltozása az *absztrakt esemény*. Minden állapot és tulajdonság változása tehát esemény. Az entitás ábrázolásakor esetleg el is tekinthetünk bizonyos tényleges állapotok ábrázolásától, mindössze az esemény bekövetkeztét jelentjük ki.

Az entitás adott eseményei annak adott kapcsolatain keresztül terjedhetnek és hatást válthatnak ki a többi entitásban.

Az entitás állapotát megváltoztató módszer a *transzformáció* vagy más néven *módosító*. Az entitás valamely tulajdonságának beállítása, vagy egy állapot „felvétele” valójában transzformáció. Ugyancsak transzformáció a kapcsolatok, illetve a kapcsolódó entitások kiépítése és lebontása. Ilyen elemi transzformációk együtteseiből összetett transzformációk képezhetők.

A transzformációk segítségével a külső entitások hatást gyakorolhatnak az entításra. Az összekapcsolódás alapvető módja, hogy adott entitás adott eseménye – az együttes környezetük absztrakt állapotának figyelembevételével – kiváltja az entitás valamely transzformációját.

Az entitás a hatás/reagálás mechanizmusok mellett adott tevékenységet (önálló változást) is végrehajthat. A tevékenység egy önmagára kifejtett hatásként is tekinthető, amely időben valamikor később, valamely következő időpillanatban jelenthet változást. A tevékenységek a következőkből épülhetnek fel:

- a környezet állapotától függően
 - belső transzformáció, vagy
 - akció, amely vagy
 - külső elem transzformációja (közvetlen), vagy
 - eseménygenerálás (közvetett)
 - reakció a külső vagy belső eseményekre
- várakozás

Korábban nem tárgyaltuk, de az entitás által végrehajtott minden tevékenység valójában egy-egy alentitásként tekinthető (!), így rendelkezhet állapotokkal/tulajdonságokkal, azonosítóval, eseményekkel, transzformációkkal, illetve altevékenységekkel...

Egyetlen entitást így a következő elemekkel adhatunk meg:

- Azonosító
- Absztrakt állapotok (alap és származtatott állapotok)
- Kapcsolatok
- Események
- Transzformációk
- Tevékenységek

A tipizálások segítségével meghatározhatjuk bizonyos entitások közös jellegzetességeit, és azokat egy-egy *típusba* emelhetjük ki.

Az entitások megadásakor elegendő azok típusaira (egy entitás természetesen több típusba is tartozhat) visszahivatkozni, s csak az entitásra vonatkozó eltéréseket vagy kiegészítéseket kell rögzíteni.

Az entitások a kapcsolataikon keresztül közvetlenül is hathatnak egymásra, illetve felvehetik egymás hatásait.

Az összekapcsolás egy nagy változatosságot biztosító, *közvetett* módja a környezetek ábrázolása.

A *környezet* ugyancsak entitás jellegű, így lehetnek állapotai, eseményei, stb. Azonban a környezet (mint szemléleti mód) annyiban több, hogy az

tartalmaz bizonyos entitásokat, valamint arra vonatkozó szabályokat. A szabályok például meghatározhatják a hatások terjedését („közvetítő közeg” jelleg), azok esetleges módosulását, a környezet állapotainak figyelembe vételével. Bizonyos szabályok entitások közötti viszonyokat írhatnak le; ezek (mint logikai értékű függvények) valójában a környezet absztrakt állapotainak, mint bizonyos entitások összevetett állapotainak a megfogalmazására szolgálnak.

Környezetek ábrázolása esetén az entitás

- beléphet egy vagy több környezetbe, illetve kiléphet környezetekből
- a környezet felé elérhetővé teheti bizonyos absztrakt állapotait, illetve felhasználhatja a környezet állapotait.
- a környezet felé továbbíthatja bizonyos eseményeit, illetve reagálhat azokra.
- engedélyezheti, hogy a környezet végrehajtsa bizonyos transzformációit

A környezet mint makroentitás ugyancsak végrehajthat bizonyos tevékenységeket, amelyek több entitás részvételét is igényelhetik.

A fentebb felsorolt, alapvetően egyszerű eszközökkel ábrázolható egy rendszer, azaz egy absztrakt változás. Az ábrázolás alapvető módja az eszközkészlettel a rendszer elemeire az egyenként és az együttesen vonatkozó szabályok megfogalmazása. A tételek segítségével a rendszer belső változásának a szabályai akár automatikusan is meghatározhatók (ismeretszerzés), olyan értelemben, hogy különböző modellek készíthetők, amelyek érvényesége ellenőrizhető.

(Jelen megközelítésünkben nem térünk ki erre, csak megemlíjtük, hogy a szabályosság keresése alkalmazható a külvilág akcióira „jobb” reakciókkal „válaszoló” szabályok kialakítására (tanulás), azaz a szabályok módosítására, vagyis evolúciós rendszerek kialakítására.)



Objektumorientált szemlélet

Az objektumorientált módszertanok történetében a szemléleti kérdések szempontjából az OMT tekinthető az egyik csúcspontnak. Megjelenésének és elterjedésének éveivel körülbelül egyidőben lassan összegyűlt annyi gyakorlati tapasztalat, amely már kevesebb elméleti háttér segítségével is lehetővé tette a sikeres rendszerfejlesztéseket. Az azóta tartó időszakban egyre inkább az ipari használat technikai kérdései kerültek előtérbe, a szemléleti kérdések pedig időlegesen a háttérbe húzódtak. A következő néhány oldalon áttekintjük az OMT, illetve az azt követő UML legfontosabb fogalom-értelmezéseit, megadjuk a szemlélet alapján a pontosításokat, illetve esetenként a helyesbítéseket. Célunk egy áttekintő kép alkotása az objektumorientált-ság fogalomrendszeréről.

A valóság modellezése

Az összeállítók szándéka alapján az OMT egy olyan szoftverfejlesztési eljárás, amely a „valóság objektumainak modellezésén alapul”^[ix] (az index az OMT-t ismertető könyv oldalszáma utal). A könyv megközelítésében az objektumorientált technológia „több mint egy pusztán programozási módszer. A probléma egy absztrakt megközelítési módja, amely elsősorban a valóság fogalmaira, nem pedig számítógépes fogalmakra épül”^[ix]. „Az objektumorientált modellezés és tervezés így a problémákról való gondolkodás egy új módszere, amely a valóság fogalmaira épül.”^[i]

A szerzők már az első oldalakon kijelentik, hogy „egy informális megközelítést” alkalmaznak „a könyvben; nincsenek bizonyítások és formális definíciók görög betűkkel, helyette ötleteket és példákat adunk a jó és rossz tervezésre”^[x]. A könyv céljához természetesen nincs szükség nagyobb formalításra. Ennek hiánya mindössze kisebb pontatlanságokban jelentkezik, illetve néhány összefüggés a definíciók informalitása miatt nem vehető

észre. Magának a jelölésrendszernek és módszernek a használatához azonban mindezek nem is szükségesek.

Az objektumorientált módszerek megjelenésével lehetővé vált az elvek nagyipari gyakorlatban történő alkalmazása. Az összegyűlt tapasztalat napjainkban már elegendő ahhoz, hogy a fejlesztés leggyakoribb problémáira kipróbált megoldásokat adjon. Az alkalmazás vázát előállító speciális, régebben rendszerprogramozásnak, újabban inkább architektúrális programozásnak nevezett területeken azonban csak a megfelelő szemléleti háttér birtokában lehet a felmerülő kérdésekre a megfelelő válaszokat adni. Ezeken a területeken ugyanis rendkívül gyakran jelennek meg a korábbi esetekhez alig hasonlító, új problémák. Ugyancsak lehetséges, hogy éppen a megfelelő formális háttér hiánya okolható azért, hogy az utóbbi években nem történt ugrásszerű, lényegi változás sem a nyelvek, sem a jelölések, sem a módszertanok területén.

Absztrakció és az OMT modelljei

Az OMT-t ismertető könyvben nagyon gyakran jelenik meg az *absztrakció* fogalma. Mint korábban láttuk, maguk a szerzők az objektumorientált technológiát is „a probléma egy absztrakt megközelítési módjaként”^[ix] értelmezik. Ebben az esetben ez azt jelenti, hogy a feladat megfogalmazásakor nem a megoldás számítógépes technikáira helyezjük a hangsúlyt, hanem először magát a problémát kísérjük meg leírni a valóságon alapuló fogalmakkal. Itt természetesen felmerül a kérdés, hogy hogyan írható le a valóság, illetve ahhoz milyen eszközöket célszerű használnunk. Az OMT, a többi rendszer-szervezési jelölésrendszerhez hasonlóan ezért a valóság leírásának kísérleteként is felfogható.

A könyvben megkülönböztetik az absztrakció fogalmát – ami szószerint „elvonatkoztatást” jelent – az általánosítás fogalmától, ami itt adott, specializáltabb elemek közös jellemzőit kiemelő általános elemként jelenik meg. A rendszerfejlesztésben megjelenő absztrakciót úgy értelmezhetjük, hogy a feladat átgondolásakor elvonatkoztatunk a későbbi informatikai megoldási módtól. Először a feladat egy absztrakt, elvi leírását készítjük el, amely kezdetben sokkal áttekinthetőbb és könnyebben módosítható a már kész programrészeknél.

Néhány oldallal később megtaláljuk az absztrakció egy értelmezését. „Az absztrakció fogalma lefedi azt, hogy az entitás lényeges, jellemző szempontjaira összpontosítunk és figyelmen kívül hagyjuk az eseti tulajdonságokat. A rendszerfejlesztésben ez azt jelenti, hogy arra összpontosítunk, hogy melyik objektum mit csinál, mielőtt eldöntenénk, hogy azt hogyan implementáljuk.”^[7] Azaz itt először egy elvi szerkezetet és egy elvi működést határozzunk meg a tényleges implementáció előtt.

Néhány oldallal később a következő definíciót találjuk: „absztrakció az egy probléma adott szempontjainak szelektív vizsgálata”^[16]. Mindez azt jelenti, hogy azt, hogy egy tulajdonságot „lényegesnek” vagy „esetinek” tartunk, azt a szempont határozza meg, illetve a problémának különböző szempontokból különböző képeit rajzolhatjuk meg.

Szemléletünk alapján a probléma vizsgálatát ezen fogalmak alapján úgy értelmezhetjük, hogy így egyrészt meghatározunk egy szempont-csoportot, amelyek együttesen egy összetett szempontot alkotnak, majd a problémára, illetve annak elemeire előállítjuk a csoportba tartozó szempontoknak megfelelő képek egyesítéséből előálló összetett képet.

Az elvonatkoztatások eredményeként előálló kép egy általános kép, ezért az absztrakció fogalma nagyon közel áll az általánosításhoz. A különbséget megérthetjük például az „absztrakt megoldás” és az „általános megoldás” fogalmak összehasonlításával. Az általános elem a változatok gyűjtőfogalma, az absztrakt jelzőt pedig általában egy elvi szinten megjelenő elemre alkalmazzuk.

Az objektumorientáltságban egy ugyancsak gyakran használt fogalom a *bezárás*. „A bezárás (más néven: információrejtés) fogalma lefedi az objektumnak a más objektumok által elérhető külső szempontjainak leválasztását a belső implementációs részletektől, amelyek rejtettek a többi objektum elől.”^[7] A definíció miatt a bezárás eredménye az objektum egy absztrakt képe lesz, mivel elvonatkoztatunk a megoldási módtól.

A rendszerfejlesztés során rendkívül lényeges szempont a rendszer felbonthatósága kisebb részekre, mivel így a figyelmünket sokkal könnyebb a teljes rendszer helyett a részekre összpontosítani. Az aktuális felbontás használhatóságát, előnyösségét pontosan az határozza meg, hogy egy-egy rész elkészítésekor mennyire kell más részeket is figyelmünk körébe vonnunk. A bezárás lehetővé teszi a részek közötti függőségek csökkentését. „A bezárás megakadályozza a program olyan kölcsönös összefüggéseit, hogy egy kis változás hatása továbbgyűrűzzön a program nagy területeire.”^[7]

A következetesen alkalmazott bezárás technikájának segítségével azt is elérhetjük, hogy annak eredménye az objektum egy elvi képét határozza meg. Az objektumorientáltság egyik nagy újdonsága, hogy programnyelvi szinten is elérhetővé tette a bezárást. Ez azt jelenti, hogy az elvi, absztrakt kép már részben magában a programban is megjelenhet, ami megkönnyíti a rendszer elkészítését és áttekintését, így például a rendszer más részeiből kizárólag ezen az elvi képen keresztül engedélyezzük az objektum elérését.

Az OMT céljainak ismertetésekor ugyancsak alapvető szerepet kap a *modellezés*, illetve a *modell* fogalma. A szerzők kijelentése szerint az OMT a „valóság objektumainak a modellezésén alapul”^[ix]. A modell fogalmára sajnos csak egy nagyon vázlatos definíciót kapunk: „A modell az valaminek az absztrakciója, hogy megérthessük azt, annak elkészítése előtt.”^[15] Tehát itt a modell is egy absztrakció, bár sajnos nincsenek felsorolva a többi absztrakciótól megkülönböztető jellemzői.

Az OMT a kifejlesztendő rendszer modellezésére három modell elkészítését ajánlja. Mindhárom modell alapvetően grafikus diagramokból áll, melyek a modellnek megfelelő elemeket és az azok közötti viszonyokat rajzolják meg.

A rendszer strukturális szerkezetét az OMT az *objektummodellel* írja le, mely elsősorban osztálydiagramokból áll. Ebben a főszereplők az objektumok, illetve ezek típusleírásai, az *osztályok*, melyeket az attribútumaikkal és műveleteikkel adunk meg. Az osztályok között viszonyokat is ábrázolhatunk, így a *kapcsolatot*, valamint adott osztályok és általános változataik közötti *általános-pontosítás viszonyt*.

A *dinamikus modell* az időbeli változást írja le, mely elsősorban az egyes osztályokra elkészített állapot-átmenet diagramokat jelent. Ezen a diagramtípuson a főszereplők a kiválasztott objektumnak az állapotai, melyeket átmenetekkel kapcsolunk össze. A diagram megadja, hogy adott állapotú objektum a külső hatásokra hogyan reagál, például hogyan változik.

Az OMT harmadik modellje, a *funkcionális modell* bizonyos értékek függését határozza meg, melyet adatfolyam-diagramokon ábrázolt számítási eljárásokkal és az azok között áramló adatokkal rajzolunk meg.

Objektum: attribútumok és műveletek

Az objektumorientáltság központi fogalma maga az *objektum*. „Az objektumot fogalomként, absztrakcióként vagy dologként definiáljuk, mely határozott körvonalakkal és a probléma szempontjából vett határozott jelentéssel rendelkezik.”^[21] Az objektum tehát gyakorlatilag bármi, ami a környezetétől megkülönböztethető és leválasztható, így a korábbi fejlesztési módszerek „*entitás*”, vagy más néven „*egyed*” fogalmához áll közel. Az entitások a rendszer többé-kevésbé különálló, így részben függetlenül vizsgálható és egymástól megkülönböztethető részei. Az egyediség egy bizonyos körülrajzolhatóságot jelent, a környezettől való elválaszthatóságot, képletesen: a valóság csomópontjait. Érdekes, hogy az OMT az értelmező definíciójában inkább egy technikai szempontból közelít: „Az egyediség azt jelenti, hogy az adatokat diszkrét, megkülönböztethető egyedekhez rendeljük, amelyeket objektumoknak nevezünk”^[2], azaz a teljes rendszer minden adatát annál az objektumnál adjuk meg, amelyre az adat vonatkozik.

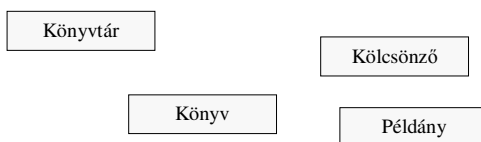
A korábbi fejlesztési módszerek egyed fogalmától az objektum fogalma abban is eltér, hogy nem csak az objektumra jellemző adatokat, a szóhasználat szerint: attribútum-értékeket adhatjuk meg, hanem az objektumokkal kapcsolatos műveletek is.

A legtöbb objektumorientált nyelvben, illetve fejlesztési módszerben mégsem objektumokat, hanem *osztályokat* definiálunk. Az osztály fogalma a szemléletben annyira alapvető, hogy az „objektumorientált” elnevezés helyett jogosabb lenne az „osztályorientált” jelző, mivel a legtöbb eszköz csak olyan objektum elkészítését engedélyezi, amelynek már definiáltuk az osztályát.

Az osztály valójában az objektumok típusa. Az OMT első definíciója sajnos semmitmondó: az osztály „egy absztrakció, amely az alkalmazás szempontjából lényeges tulajdonságokat leírja és a többit figyelmen kívül hagyja”^[2]. A definíciót úgy értelmezhetjük, hogy az osztály a rendszer olyan fogalmait reprezentálja, amely fogalmak alá, mint típusok alá rendelhetők az objektumok. Ez a hozzárendelés az osztályozás. „Minden objektumról azt mondjuk, hogy az az osztályának a *példánya*”^[2].

Az OMT-ben az objektumot néha kettős értelemben használják. Az *objektumpéldány* jelenti a konkrét példányt, az *objektumosztály* pedig az osztály típus-jellegű fogalmát. Ma már egyszerűen csak az osztály megnevezést

alkalmazzák, az „objektum” elnevezésen pedig mindig objektumpéldányt értenek. A későbbi definíció már részletesebb: „Az objektumosztály az objektumok olyan csoportját adja meg, amelyek hasonló tulajdonságokkal (attribútumokkal), közös viselkedéssel (műveletekkel), közös, más objektumokhoz való viszonyokkal, valamint közös szemantikával rendelkeznek”^[22]. Ehhez hasonló az osztályozás fogalma is: az „osztályozás azt jelenti, hogy az azonos adatstruktúrával (*attribútumokkal*) és viselkedéssel (*műveletekkel*) rendelkező objektumokat osztályokba csoportosítjuk”^[2]. Az osztályozás célja tehát, hogy ne csak különálló objektumokat tudjunk megadni, hanem azok közös jellegzetességeit ismétlés nélkül, egyetlen helyen sorolhassuk fel. „Az objektumok osztályokba való csoportosításával absztrakt módon közelelíthetjük meg a problémát”^[22].



36. ábra. Osztályok

A központi szerepű objektummodell alapvetően osztálydiagramokat tartalmaz, így „az objektummodell a rendszer statikus struktúráját rajzolja meg, megmutatva a rendszer objektumait, az objektumok közötti viszonyokat, valamint az objektumok osztályait jellemző attribútumokat és műveleteket”^[21].

Az elnevezés mellett az osztály megadásának egyik eszköze az attribútumainak a felsorolása. Az OMT leírásában néha összekeveredik az attribútum és az attribútum-érték fogalma, mint például a következő definícióban: „az attribútum az osztály objektumában tárolt érték”^[23].

Osztály esetén tehát a nevükkel és típusukkal attribútumokat adunk meg, melyhez az objektumok más-más értéket rendelhetnek. „Az osztály minden példánya minden attribútum esetén saját értékkel rendelkezik, de az attribútum-neveken és a műveleteken az osztály többi példányával közösen osztoznak.”^[2]

Sajnos pontosabb definíciót nem találunk, csak még egy kijelentést: „az attribútum csak egy egyszerű érték lehet, nem pedig objektum”^[23]. A ké-

sőbbi módszerek pontosították ezt a kijelentést, így összetett értékeket (mint például egy dátum, vagy egy cím) objektumokba csoportosíthatunk, melyeket érték-objektumoknak nevezünk, szemben a valóság entitásait reprezentáló entitás-objektumokkal.

Szemléletünk alapján pontosíthatjuk a definíciót. Az attribútum itt az objektum adott szempontból vett tulajdonsága. Az *attribútum-név* (például egy *személy* objektum esetén a *születési idő* attribútum-név) elnevezi a szempontot, amelynek konkrét objektum esetén konkrét *attribútum-értéke* lesz (a születés konkrét dátuma). Az *attribútum típusa* a lehetséges értékeket (azok halmazát) határozza meg (születési idő esetén a dátumok halmaza). Az értéket pontosan az különbözteti meg az entitástól, hogy az érték a lehetőségek halmazából index-jellegű módon választ ki egy lehetőséget, így nem identitás, hanem csak szelektor szerepű. Az objektum absztrakcióját eredményezi annak az attribútum-névvel elnevezett szempont szerinti képe, amely a szempont és az érték együttese lesz (pl. a szélesség méterben: 2).

Az attribútumok valójában azok az eszközök, melyekkel a teljes objektum együttes állapotát az egyes szempontok szerint, elemenként, azaz a szempont által meghatározott lehetőségek közül választásként adhatjuk meg.

Az objektum osztálya az objektum absztrakt képe, mivel eltekintünk a ténylegesen felvett attribútum-értékeiktől. Ha az objektumot egyetlen attribútumára szűkítjük, akkor ugyancsak az absztrakcióját, az attribútum-név által meghatározott *szempontból* vett képét kapjuk.

Alakzat
hely: Location
méret: Size
kirajzolás ()
mozgítás (újHely: Location)
nagyítás (faktor: Real)
forgatás ()

37. ábra: Attribútumok és műveletek

Az objektumorientált rendszerekben az objektumok egyediségének technikai megoldása az *objektumazonosító*. Egy objektumorientált rendszerben akkor érhetünk el egy objektumot, ha rendelkezünk annak az objektumazonosítójával.

Az entitásnak így az absztrakciója az egyedisége, amely azt jelenti, hogy elvonatkoztatunk minden jellemzőtől, és csak arra korlátozzuk az

információkat, hogy két egyediség – a klasszikus példa alapján két *ugyanolyan* alma – esetén *ugyanarról* vagy különböző entitásokról van szó.

Az objektumazonosítót így nem szabad attribútumként felvennünk^[24], bár az az objektum szelektora, azonban logikailag nem értékjellegű, hanem éppen az egyediség reprezentációja. A fejlesztés gyakorlatában egy érték-objektum és egy entitás-objektum közötti különbség pontosan úgy jelenik meg, hogy míg entitás-objektumok esetén az *azonosságot* (az objektumazonosítók egyediségét) ellenőrizzük, addig érték-objektumok esetén az *egyenlőséget*, mégpedig az attribútum-értékek együttes egyenlőségét.

Az osztálydiagramon az attribútumokat követően felsorolhatjuk az osztályokhoz tartozó műveleteket. Az OMT definíciója itt is nagyon vázlatos: „a valóságban a művelet egyszerűen a különböző fajtájú objektumok hasonló viselkedésének az absztrakciója”^[3].

Valamivel többet mondanak az alváltozatok definíciói: „a *művelet* egy akció vagy egy transzformáció, amelyet az objektum végrehajt, vagy amelyet az objektumon végrehajtanak”^[2], illetve kicsit később: „a művelet egy függvény vagy egy transzformáció, amelyet az osztály objektumaira lehet alkalmazni”^[25]. Az utóbbiak alapján az említett *függvény* fogalmán egy lekérdezést értenek, amely „kiszámít egy funkcionális értéket az objektum módosítása nélkül”^[26]. Az utóbbi definícióban a „módosítás nélkül” már pontosítódik: „a lekérdezés egy olyan művelet, amelynek nincs mellékhatása egyetlen objektum kívülről látható állapotára sem”^[131]. Az akcióra a következő definíciót találjuk: „az akció egy olyan transzformáció, amelynek van mellékhatása a célobjektumra vagy a rendszernek a célobjektumtól elérhető további objektumaira”^[131]. Ezen értelmezés alapján a korábbi definícióban az „akció vagy transzformáció” valószínűleg egyszerűen elírás.

A definíciókban itt is sok a hiányosság, amelyet a szemléletünkkel pontosítani lehet, sőt, a legtöbb pontosítás magában az OMT fogalomkörében is ellenőrizhető. Lekérdezés és transzformáció mellett megadható például olyan művelet, amely nem csinál semmit, vagy azért mert nincs benne műveletsorozat, vagy azért mert az éppen adott feltételek mellett nem módosul az objektum. Így helyesebb lenne a transzformáció *lehetőségét* hangsúlyozni, vagy megjegyezni, hogy a transzformáció lehet identikus is, azaz az eredeti állapotot is megadhatja. Ugyancsak lehetséges, hogy egy művelet közvetve, valamely más objektumon keresztül fejt ki hatást. Ilyenek például a napjainkban gyakran használatos

„jelentő” vagy „tüzelő” metódusok, amelyek nem módosítják az objektumot, hanem éppen a módosítás szükségességét jelzik a vele kapcsolatban lévő objektumok felé.

A szemléletben a műveletek a hatások átvitelének a megadási lehetőségei. A hatás úgy értelmezhető, hogy a ható objektum állapota „adott mederbe tereli”, *korlátozza* a másik, a „függő” objektum által felvehető állapotot. Mivel informatikai környezetben vagyunk, a hatás átvitelét a vezérléssel rendelkező, ún. aktív objektum kezdeményezheti. Ha a ható objektumnál van a vezérlés, akkor az a függő objektumnak egy transzformáció-jellegű metódusát meghívja és ezen keresztül közli vele, hogy milyen állapotba kell kerülnie. Ha a függő objektumnál van a vezérlés, akkor annak le kell kérdeznie a ható objektum állapotát, és attól függően kell módosítania a saját állapotán. Természetesen, ha a függő objektum már a megfelelő állapotban van, akkor nem történik transzformáció. A „jelentő” metódusok valójában egy indirekciót tesznek lehetővé a hatás átvitelében.

Az OMT a *művelet*en valójában egy művelet-jelleget ért, mint az objektumok „hasonló viselkedésének az absztrakciója”^[3]. „Adott osztály műveletének a specifikus megvalósítását *metódusnak* nevezzük”^[2], tehát azt a pontos lépéssorozatot és számításokat, amely a nevével jelölt absztrakt műveletet adott módon valósítja meg. Például egy *forgatás* megnevezhet egy adott jellegű műveletet, és más utasítássorozat szükséges például egy téglalap elforgatásához, és más egy kép esetén.

Az objektumorientáltságban megadhatunk azonos nevű műveleteket, amelyek csak a paramétereik számában és/vagy típusában különböznek, például egy racionális törtszámhoz hozzáadhatunk tört- és egészértéket is.

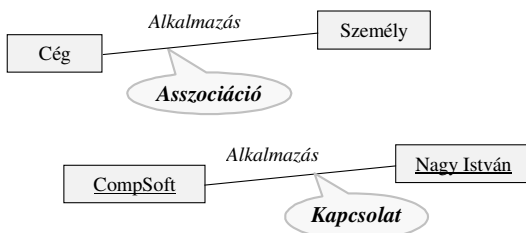
Mivel a név a művelet jellegére utal, ezért a konkrét utasítássorozattal megadott metódusnak az absztrakciója lesz a hívási módot meghatározó név és paraméterek, ennek pedig az absztrakciója lesz a hívási módtól is elvonatkoztató, csupán a nevével jelölt művelet-jelleg.

Osztályok közötti viszonyok

Rendszerünk objektumai között különböző viszonyok lehetségesek. Közvetlen viszonyt jelent például az elemek összekapcsolódása. A klasszikus példa szerint egy személy alkalmazási kapcsolatban van az őt alkalmazó céggel.

Az OMT különböző elnevezéseket alkalmaz az objektumok, azaz a konkrét példányok közötti tényleges kapcsolatra, és a típusok, azaz az osztályok közötti viszonyra, melyet asszociációnak nevez. „A kapcsolat egy fizikai vagy fogalmi kapcsolódás az objektumpéldányok között.”^[27] „Matematikailag a kapcsolat egy elem n -es, azaz adott számú” (n elemű) „rendezett listája az objektum-példányoknak.”^[27] „Az asszociáció a közös szerkezetű és közös szemantikájú kapcsolatok csoportját határozza meg.”^[27]

Mindezt úgy is mondhatjuk, hogy az asszociáció a kapcsolat típusa, így azt osztályok között adhatjuk meg. Ezért egyrészt „az asszociáció a lehetséges kapcsolatok csoportját határozza meg, hasonlóan, ahogy az osztály a lehetséges objektumok halmazát határozza meg”^[27], másrészt pedig „a kapcsolat az asszociáció példánya”^[27], tehát itt is egy absztrakciós viszonyt találunk.



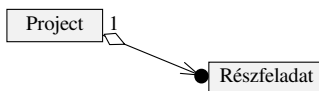
38. ábra. Asszociáció és kapcsolat

„A szerepkör az asszociáció egyik vége”^[34], például az alkalmazás kapcsolatban a szerepköröket elnevezhetjük „alkalmazó” és „alkalmazott” módokon. A szerepköröknél jelölhető a *multiplicitás*, más néven számosság, mely „megadja, hogy egy osztály hány példánya kapcsolódhat az asszociált osztály egyetlen példányához”^[30]. A számosságot a nem-negatív számok halmazával adjuk meg, például egyetlen elemmel, vagy az elemek tartományával, vagy ezek felsorolásával. A legáltalánosabb a fekete körrel (az UML-ben * jellel) jelölt *tetszőleges számosság*, mely azt jelenti, hogy egy objektumhoz az asszociációnak megfelelő típusú, tetszőleges számú objektum kapcsolódhat.

Itt is megjelenik az általánosítás, mivel egy számosság pontosítható úgy, ha a nem-negatív értékeknek a számosság által megadott halmazának egy részhalmazát választjuk ki. Például a tetszőleges számosságot jelentő teljes halmaznak a legfeljebb 10 (értsd: [0;10] intervallum) a részhalmaza, annak az opcionális jelölést jelentő 0 vagy 1 érték (értsd: [0;1])

a részhalmaz, annak pedig a „pontosan egy” egyelemű halmaz ugyan-csak a része.

Az OMT külön is tárgyalja az asszociáció egy speciális változatát, az aggregációt, mely egy elem és annak alkotóelemei közötti viszonyt ír le. „Az aggregáció egy *rész-egész* vagy *része* viszony, amelyben a valaminek a komponenseit reprezentáló objektumok kapcsolódnak az egészt reprezentáló objektumhoz.”^[36] „Az aggregáció az asszociáció erősebb formája, amikor az aggregált objektum a komponensekből épül fel.”^[57]



39. ábra: Aggregáció

Az aggregáció külön tárgyalását az indokolja, hogy az ilyen jellegű objektum-kapcsolatok általában az objektumok speciális együttműködését, interakcióját is jelentik. „Az aggregáció jellemző tulajdonsága a tranzitivitás” (tömbház része a lakás, lakásnak a szoba, így a tömbház része a szoba), az „antiszimmetria” (a lakás nem lehet a szoba része), valamint a „propagáció”,^[37] mely utóbbi azt jelenti, hogy az „egész” objektum bizonyos attribútum-értékeit felajánlhatja a részeinek, melyeket azok alapértelmezésként tekintenek, de meg is másíthatnak. Például egy cég értesítési címe alapértelmezésként a telephelyeinek is az értesítési címe, de az módosulhat a telephely címére is. Ilyenkor, ha a *résznél* nem definiált a speciális érték, akkor az az *egész* megfelelő értékét fogja alkalmazni.

Hasonló interakció-típus a műveletek delegációja. „Az aggregátum szemantikailag egy kiterjesztett objektum, amely több műveletben egyetlen egységként jelenik meg, bár fizikailag több kisebb objektumból épül fel.”^[57] A programban mindez általában úgy vehető észre, hogy „az egész bizonyos műveleteit automatikusan alkalmazni kell a részekre is”^[58]. Például egy dokumentum kinyomtatásakor annak minden egyes lapja is kinyomtatódik, vagy egy ház lebontása során annak lakásai is megszűnnek. Pontosán ez utóbbi kapcsolat nagyon gyakori: „a komponens-objektum létezése függhet az aggregátum-objektum létezésétől”^[38]. Az élettartamok függősége nagyon sok kritikát kapott, mivel pl. egy cég megszűnése szerencsére nem jelenti automatikusan az alkalmazott személyek megszűnését is. Az OMT utóda-

ként is tekinthető UML-ben ezért az aggregációt csak a rész-egész viszony jelzésére alkalmazzák, mindenféle külön szemantika nélkül, valamint bevették annak erősebb változatát, a kompozíciót. A kompozit objektum a komponenseiből épül fel és a szlogen szerint azokkal „együtt él és hal”. Ekkor egy komponens, mint építőelem csak pontosan egy kompozíthoz tartozhat.

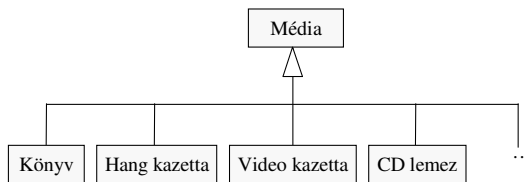
A kapcsolatokról és asszociációkról sajnos nem találunk több definíciót. Szemléletünk értelmezésében a *kapcsolat* a hatások átvitelének az útja, a kapcsolódó objektumok tehát így tudnak egymásra hatást kifejteni. A *szerepkör* a kapcsolódó objektum absztrakt képe, mivel azt eleendő csak a hatás átvitelének szemszögéből tekinteni, a többi jellemzőjétől pedig elvonatkoztathatunk. Az UML már külön jelölést is alkalmaz a szűkített módon történő kapcsolódás ábrázolására.

Érdemes megfigyelni, hogy a kapcsolat maga is egy kiterjesztett objektum, amely az aggregációhoz hasonlóan „több műveletben egyetlen egységként jelenik meg, bár fizikailag több kisebb objektumból épül fel”, csak kapcsolat esetén nem feltétlenül aszimmetrikus a viszony, valamint nem jelöljük az egyediséget, azaz a kapcsolat általában nem jelenik meg külön objektumként.

A hatások alapján az aggregáció, illetve kompozíció fogalmai is könnyen értelmezhetők. Az *egész* hatását a *részekre* részben a propagáció írja le, illetve a műveletek továbbításával, delegációjával az egészt ért hatás közvetve a részekre is hatást gyakorolhat.

A kapcsolat és az asszociáció mellett az OMT definiálja az osztályok közötti általánosítás-pontosítás viszonyt. „Az általánosítás egy viszony az osztály és annak egy vagy több pontosított változata között.”^[39] „Az általánosítás és származtatás a hasonlóságoknak az osztályok közötti megosztásának erőteljes absztrakciós eszközei, miközben azok megőrizhetik a különbségeiket.”^[38]

Az osztályok az *azonos* attribútumokkal és műveletekkel rendelkező objektumoknak az absztrakt változatai. Az általános osztály, más néven szuperosztály alkalmas arra, hogy a több osztályban előforduló *közös* attribútumokat, műveleteket és asszociációkat kiemeljük, és így egyetlen helyen definiáljuk. Természetesen csak akkor célszerű ez a kiemelés, ha a közös jellegzetességekkel rendelkező osztályok a valóságban is az általános szuperosztály egy-egy változatát reprezentálják.

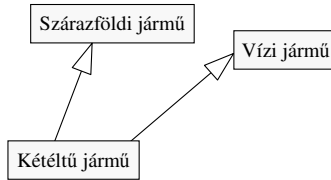


40. ábra: Általánosítás

Az objektumorientált rendszerek a változatoknak az általános elemhez történő hozzákapcsolását általában az öröklés technikájával valósítják meg. Az öröklés során az alosztályok átveszik, „öröklik” az ősosztályban definiált jellemzőket, így annak attribútumait, műveleteit és asszociációit, melyet közösen jellegzetességeknek nevezünk. „Az öröklés az attribútumoknak és műveleteknek az osztályok közötti megosztása egy hierarchikus viszony alapján.”^[3] A hierarchikus viszonyt az indokolja, hogy „az általánosítás és az öröklés tranzitív, tetszőleges számú szint esetén”^[39].

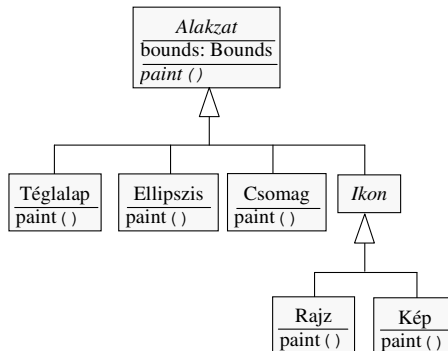
Az általánosítás-pontosítás elvi szinten megfelel az „az-egy” (az angolban: „is-a”) viszonynak^[39], amely leggyakoribb megvalósítási módja az öröklés. A viszony azt jelzi, hogy több fogalom mögött találtunk egy általánosabb fogalmat, például a repülőgép, autó és hajó egyaránt jármű. A korábbi idézet alapján ez is egy absztrakciós eszköz, mivel esetinek tekinti az objektumok jellegzetességeit és csak a közösetek tekinti lényeginek, például mindegyik járműnek lehetnek a sebességet, illetve a szállítható utasokat meghatározó attribútumai, valamint az elindulás, megállás, stb. műveletei. Az általános osztály segítségével a közös jellemzők kiemelhetők, illetve az objektumok ennek segítségével általánosan („járműként”) közösen kezelhetők.

Egy fogalomnak természetesen több általános változata is lehetséges, ami a többszörös öröklést jelenti. „Többszörös öröklés” esetén „egy osztálynak több mint egy ősosztálya lehet és mindegyik szülőjétől örökli jellegzetességeiket”^[65].



41. ábra: Többszörös öröklés

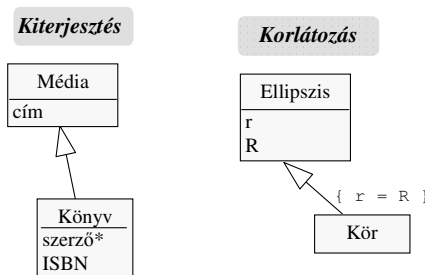
Az általánosítás során megjelenhetnek olyan osztályok, amelyeknél bizonyos műveleteket nem tudunk megadni, hanem csak kijelentjük azok létezését. Például egy valós függvény osztály esetén kijelenthetjük, hogy annak az f művelete a paraméterként kapott x értékhez megadja a hozzárendelt értéket, de ha ezt a hozzárendelést megadnánk, akkor máris egy konkrét függvényt kapnánk. Az OMT az ilyen deklaratív jellegű osztályt absztrakt osztálynak nevezi és dőltbetűs elnevezéssel jelöli; ez „olyan osztály, amelynek nincsenek közvetlen példányai, de amely leszármazottainak vannak direkt példányai”^[61].



42. ábra: Absztrakt osztály

Például egy lapra alakzatokat rajzolhatunk. Az absztrakt alakzat osztálynál csak kijelentjük, hogy minden alakzat objektum esetén kell léteznie a rajzoló műveletnek, de ha azt megadnánk, akkor máris egy konkrét alakzatot kapnánk. Viszont csak olyan osztálynak készíthető el objektuma, amely ehhez a művelethez metódusként konkrét utasítássorozatot rendel.

Az alváltozatok megadásakor két eszközünk is van. *Kiterjesztéssel* „az alosztályhoz új jellegzetességeket adunk”^[63]. A másik eszköz a *korlátozás*, mely segítségével „az alosztály korlátozhatja az ős attribútumait”^[63].



43. ábra: Kiterjesztés és korlátozás

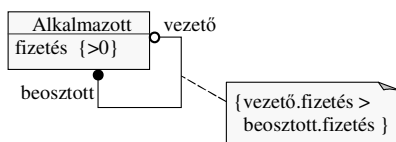
Az OMT-könyvben a szerzők közismertnek tekintik az általánosítás és pontosítás fogalmait. Mint láttuk, az osztályok esetén alkalmazott általánosítás is tekinthető absztrakcióként, mivel kiválogatja a szuperosztály nevével jelölt szempont szerint az alosztály vagy alosztályok adott jellegzetességeit. Ezen absztrakció eredményeként nem egy, az eredeti objektumtól elszakított értéket kapunk eredményül (ahogy attribútumok esetén), hanem az objektum általános módon, csak a kiválasztott jellegzetességeivel történő kezelését tesszük lehetővé.

Szemléletünkben a hatások alapján az elemek általánosítása is értelmezhető, mely szerint az elemek hasonló jellegét úgy határozhatjuk meg, mint amelyek azonos hatásra azonos jellegű, bár nem feltétlenül megegyező módon reagálnak, illetve a környezetükre azonos jellegű hatással lehetnek. Például egy médiatárban a kölcsönzés „hatására” egy könyv és egy CD is hasonló módon reagálhat, bár a kölcsönzési idejük eltérhet.

Az objektumok közötti viszonyok esetén kiemelt szerepet kapnak a kapcsolatok/asszociációk, illetve az általánosítás-pontosítás viszonyok. A modell elemei között azonban létezhetnek olyan további viszonyok, amelyek a két típus közül egyiknek sem feleltethetők meg. Az OMT jelölései esetén a korlátozó megszorítások, röviden *korlátozók* (amelyet a *korlátozástól* eltérő értelemben használunk; a korlátozás a pontosítás egy változata) jelentik azt az általános eszközt, amellyel tetszőleges viszonyt is kifejezhetünk: „a korlátozók további viszonyokat határoznak meg, amelyeket fenn kell tartani az objektummodellben”^[140]. Az első definíció szerint: „a korlátozók funkcionális viszonyok az objektum-modell entitásai között”^[73]. Itt is találunk egy kis

pontatlanságot, mivel a korlátozó egyetlen elemre is vonatkozhat, ahogy azt a következő definíció is mutatja: „a korlátozó az entitás által felvehető értékeket korlátozhatja”^[73]. A későbbiekben egy egyesített definícióval is találkozunk: „a korlátozó egy adott időpontban két objektum közötti viszonyt határoz meg, vagy egy objektum különböző időpontokban felvett különböző értékei közötti viszonyt ír le”^[132].

A korlátozó és a korlátozás (az utóbbi mint a pontosítás egy lehetősége) fogalmai úgy kapcsolódnak össze, hogy *korlátozást* a szuperosztály objektumának attribútum-értékeire, illetve kapcsolataira vonatkozó *korlátozó* megszorítással adhatunk meg.



44. ábra: Korlátozó megszorítások

Az OMT szerzői szerint minden modellelem, illetve minden köztük ábrázolt viszony önmaga is korlátozó megszorítás, így pl. „a multiplicitás az asszociációt korlátozza”, mégpedig a kiindulási „objektumhoz kapcsolódó objektumok számát”^[74]. A korlátozó önálló jelölésével azonban olyan viszonyokat is kifejezhetünk, amelyek ábrázolására nincs külön jel.

A korlátozó egy elemet pontosan is meghatározhat, például egy attribútum-értéket egyetlen értékre korlátozhat. Általános esetben a korlátozó a lehetőségek tartományának választja ki egy részhalmazát, mely részhalmaznak valódi részhalmaznak kell lenni, különben nem beszélhetünk korlátozóról. „A korlátozó lehet egy teljes függvény (egy értéket pontosan meghatároz a többi érték) vagy egy parciális függvény (egy érték korlátozott, de nem pontosan meghatározott a többi által).”^[132]

Invariánsnak nevezzük azt a korlátozót, amely egy hosszabb időtartamon keresztül fennáll, többnyire az elem (az objektum, az objektumcsoport vagy a teljes rendszer) teljes élettartamára vonatkozik. „Az objektum értékei közötti, az időben folyamatosan fennálló megszorítást általában invariánsnak nevezzük.”^[133] „Az invariánsok az értékek közötti olyan függőségeket adnak meg, amelyeknek az időben fenn kell maradniuk”^[140]

Észrevehetjük, hogy az OMT a korlátozó jelölését a modell további finomítására alkalmazza. Például az alkalmazott fizetésének számértékű attribútumánál megadhatjuk, hogy az csak pozitív szám lehet, vagy hogy

a vezető fizetésének nagyobbnak kell lenni a beosztottai fizetésénél. Ezekben az esetekben a modell alapjelölései a kelleténél általánosabb rendszert határoznak meg, amelyet pontosítanunk kell, például a hibák elkerülése végett. Minden újabb modellelem, új osztály, attribútum, művelet, viszony, stb. felvétele ugyancsak pontosítja a modellt. Például az attribútum típusának a megadása szintén korlátozó, mivel a „tetszőleges értékek” halmazának egy részhalmazát határozza meg. A típus további pontosítása (a valós számok helyett pl. egész számok megadása) vagy annak egy részhalmazának kiválasztása ugyancsak korlátozó. Mivel a diagramok maguk is korlátozóként jelennek meg, gyakorlatilag az objektummodell, illetve az összeállított teljes modell is korlátozó.

A szemléletben a korlátozó központi szerepet kap, mivel kiemeli a tetszőlegességből a rendszert, illetve további pontosításokra a lehetőséget. Általános jellegénél fogva ezért a korlátozó alkalmas az *információ* (!) definíciójára. Természetesen ez a definíció nem az információelmélet mennyiségi szempontját jelenti.

Az aggregált osztály, illetve az általánosítás viszony szuperosztálya egyaránt több elemet, nevezetesen a részeket, illetve az alosztályokat csoportosítja. Aggregáció esetén a részek logikailag vagy esetleg fizikailag is (pl. az UML kompozíció fogalma esetén) a közös aggregátumon belül helyezkednek el. A szuperosztálynak az egyes alosztályok a különböző alváltozatai.

Az OMT így ismerteti a két viszony kapcsolatát: „az aggregációt gyakran *és*-viszonynak nevezik, az általánosítást pedig *vagy*-viszonynak.”^[59] Aggregáció esetén az egész ugyanis az egyik részből, *és* a másik részből, *és* a további részeiből áll... Általánosítás esetén pedig az általánosként tekintett objektum az *vagy* az egyik alváltozat objektuma, *vagy* a másik alváltozat objektuma, stb.

Az óvatos megfogalmazás a megközelítésünkkel pontosítható. *Vagy*-viszony csak az általánosítás lehet, hiszen így eltekintünk a tényleges változattól. *És*-viszony azonban nem csak aggregáció lehet, hanem ugyancsak *és*-jellegű viszony a többszörös származtatás. Például egy kételtű jármű az szárazföldi *és* vízi-jármű is. Az aggregáció *és* a többszörös származtatás így rokon fogalmak, ami a programozás gyakorlatában egyáltalán nem meglepő, mivel egy többszörös származtatás a programban implementálható aggregációval, *és* fordítva. A két viszony a valóságban a következőkben tér el: többszörös származtatás esetén a szuperfogalmak az azonos alapelem (esetünkben objektum) szempontjaiból alkotott képeket írják le, aggregáció esetén pedig különböző részeket. Tehát amíg a többszörös származtatás szuperosztályai azonos helynek írják le a különböző szempontból vett jellegzetességeit, addig aggregáció esetén a részek különböző helyeket határoznak meg. A kereszt-implementáció lehetősége, pl. egy kételtű jármű aggregációként

történi reprezentációja úgy képzelhető el, mintha azt mondanánk, hogy az objektum „ettől eddig” (ez a része) vízijármű, a másik része pedig szárazföldi jármű. Többszörös származtatás esetén pedig mindkét elem ugyanarra a részre, azaz ugyanarra a helyre vonatkozik.

Dinamikus modell

Az OMT második modellje, a dinamikus modell, a rendszer időbeli változását hivatott leírni. A dinamikus modell is alapvetően diagramokból, az ún. *állapot-átmenet diagramokból* áll.

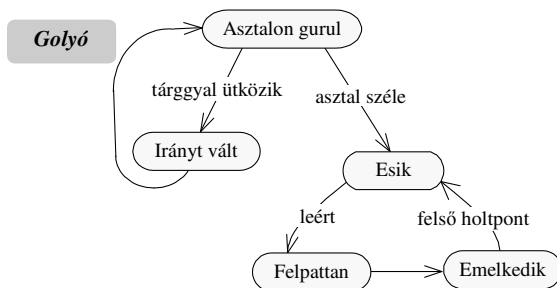
Maga az állapot-átmenet diagram egy régebbi konstrukció, amelynek az OMT-ben is használt változat alapkonceptióit David Harel határozta meg. Az OMT összeállítói a teljes jelölésrendszert átvették, és azon csak az objektumorientáltságnak megfelelő kisebb módosításokat tettek.

Az „objektum-modell leírja az objektumok, attribútumok és kapcsolatok lehetséges mintáját”^[84]. Az OMT-ben állapotdiagramot csak néhány, karakterisztikus változású osztály esetén készítenek el. Egyetlen diagramon gyakorlatilag egyetlen osztály változása szerepel, amely így sajnos csak rendkívül vázlatos képet ad a teljes rendszer változásáról.

Az OMT definíciójában a rendszer dinamikájának definiálására több kísérletet is találunk. Az első a kibernetikából ismert fogalmakat használ: „a vezérlés a rendszernek az a jellegzetessége, amely külső stimulusra”, más néven ingerre, vagy hatásra „válaszként megjelenő műveletsorozatokat írja le”^[84]. A későbbiekben használt „esemény” fogalma „a külső stimulust reprezentálja”^[84].

A többi definícióban a dinamika a változást, azaz az állapot megváltozását jelenti. Mivel a diagramot alapvetően egy osztály szempontjából készítjük el, ezért állapoton is az osztály objektumának állapotát értjük, így az állapot az „objektum értékeit reprezentálja”^[84], illetve később: „az objektum által tárolt attribútum-értékeket és kapcsolatokat együttesen állapotnak nevezzük”^[84].

Az ellentmondást néhány pontosító definícióval oldhatjuk fel: az objektum *belső állapotán* az objektum attribútum-értékeit, *külső állapotán* az objektumhoz kapcsolódó további objektumok belső állapotait értjük; az objektum *kiterjesztett állapota* így a belső és külső állapotának együtteseként definiálható.



45. ábra: Állapot, esemény és átmenet.

A következő definíció az állapotról ugyancsak egy újabb dolgot állít: „az állapot az objektum attribútum-értékeinek és kapcsolatainak absztrakciója”^[87]. Az ellentmondó megfogalmazásokat úgy oldhatjuk fel, hogy az előbbi definícióink *konkrét* külső-, belső- és kiterjesztett állapotokra vonatkoznak, amelyek *absztrakt* állapotokba csoportosíthatók. Az állapotával történő jellemzés az objektumot (illetve az ahhoz kapcsolódó többi objektumot) egyetlen egységnek tekinti, míg az attribútumok értékei az állapotot elemeire darabolják fel. Az állapot meghatározásával így egy magasabb szempontból vett, áttekinthetőbb képet adhatunk meg az objektumról, illetve annak változását is absztraktabb módon fogalmazhatjuk meg.

Az állapot az osztály általános, strukturális váza és az objektum példányszintű konkrétsága közötti absztrakciós szintet határoz meg. A konkrét állapot az objektum pontos értékeit adja meg. Az absztrakt állapot már csak korlátozóként jelenik meg, mivel „korlátozza az objektum által felvehető értékeket”^[110].

Ebben a modellben a változás a következőképpen zajlik le: „időről időre az objektumok stimulálják egymást, amely az állapotaik megváltozását eredményezi”^[84]. „Egy objektumtól egy másikra ható stimulus az esemény.”^[84] „A rendszer objektumai az események váltásával hatnak egymásra.”^[103] „Az eseményre adott válasz a fogadó objektum állapotától függ, amely tartalmazhatja az állapot megváltozását és az eredeti küldő vagy más objektum felé történő üzenetküldést.”^[84] Az „esemény hatására történő állapotváltozást átmenetnek nevezzük”^[89]. A korábbi definícióban megjelenő üzenetküldés fogalma az objektumorientáltságban az egyik objektumtól a másikig terjedő információátvitelt jelenti. A következő sor azonban az ese-

Az „állapot meghatározza az input események hatására történő választ”^[87], így megkapjuk az állapot-átmenet diagram vázát, amely azt ábrázolja, hogy adott állapotban a beérkező esemény hatására az objektum milyen állapotba vált át: „az esemény fogadásakor a következő állapot éppúgy függ az aktuális állapottól, mint az eseménytől”^[89].

Az állapotváltásnál *feltételeket* is megadhatunk, mely „feltétel az objektum értékeinek egy logikai értékű függvénye”^[91]. „A feltételek úgy alkalmazhatók, mint az átmenetek őrszemei. Az őrzött átmenet az esemény bekövetkeztekor csak akkor történik meg, ha az őrfeltétel igaz.”^[91] „Lényeges megkülönböztetni a feltételt és az eseményt, mely utóbbinak nincs időtartama”^[91], a feltétel azonban „egy adott időintervallumig érvényes”^[91].

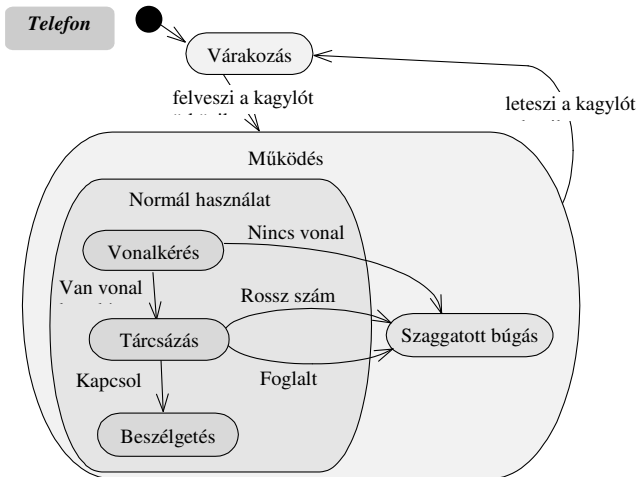
A feltétel és az állapot az OMT-ben rokon fogalmak: „az állapot gyakran hozzákapcsolható az objektum valamely értékeihez, amelyek adott feltételt teljesítenek”^[88]. Az őrfeltétel valójában annak a tesztje, hogy az objektum egy adott (kiterjesztett) állapotban van-e, mely állapotot vagy nevesítjük, vagy egy logikai értékű függvényként adjuk meg. Ezért: „az állapot definiálható egy feltétellel, illetve fordítva, az adott állapotú objektum egy feltételt jelent”^[91]. A helyzet ennél egy kicsit bonyolultabb, amely a változások modellezésének gyakorlatában úgy jelentkezik, hogy esetenként nehéz döntenünk, hogy egy átmenethez eseményt vagy feltételt írunk. Esemény megadása ekkor azt jelenti, hogy a változásról kívülről kapunk értesítést, feltétel megadása pedig azt, hogy az átmenetnél mi hajtjuk végre az ellenőrzést, amely jelentheti egy külső objektum tesztjét (!) is.

Az állapotokhoz és átmenetekhez műveleteket rendelhetünk. Például „az állapot gyakran hozzákapcsolható egy folyamatosan végrehajtott tevékenységhez” (más néven aktivitáshoz) „vagy egy olyan tevékenységhez, amelynek befejezése időt vesz igénybe”^[88], ilyen folyamatosan végrehajtott tevékenység például egy telefon csengése. A műveleteknek ez a formája, a *tevékenység* „olyan művelet, amelynek befejezése időt vesz igénybe”. Éppen ezért „a tevékenység állapothoz kapcsolható”^[92].

Az időtartam-nélkülinek tekintett „*akció* egy azonnali művelet”, így „az akció eseményhez kapcsolható”^[92]. Az OMT-ben akció pl. egy attribútum-érték megváltozása, vagy esemény küldése egy másik objektumnak.

Az állapotok és események közös jellegzetességeit az osztályokhoz hasonlóan kiemelhetjük: „az általánosítás lehetővé teszi az állapotoknak és eseményeknek az általánosítási hierarchiába történő elrendezését, a közös

szerkezet és viselkedés öröklésével, hasonlóan az attribútumok és műveletek örökléséhez osztályok esetén”^[94]. Az általánosítás egyik felhasználási lehetősége, hogy egy állapot tevékenysége „kifejthető egy alacsonyabb szintű állapotdiagramon”^[95]. Az UML jelöléseinél ez már egy külön diagram-típusként, az aktivitás-diagramként jelenik meg. A másik felhasználási lehetőség „a beágyazott állapotdiagram”, amely „az állapotok általánosításának egy formája”^[96].

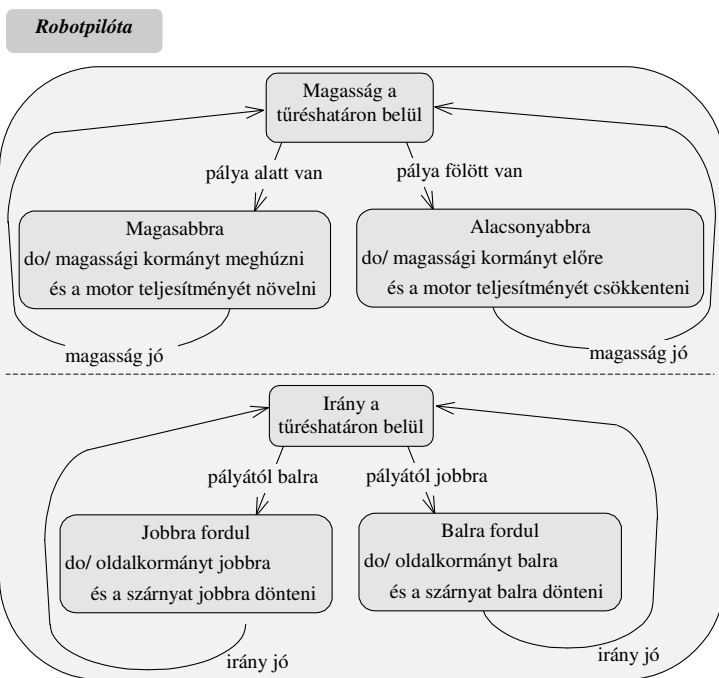


47. ábra: Strukturált állapotdiagram

Itt is megemlíthetjük, hogy „az általánosítás egy vagy-kapcsolat”, ami ebben az esetben azt jelenti, hogy „a magasabb szintű diagram egy állapotában lévő objektum, az alacsonyabb-szintű diagramnak pontosan egy állapotában van”^[96].

„Az állapotnak” így „lehetnek alállapotai, amelyek a szuperállapotaik átmeneteit öröklik”^[97], illetve változatként átdefiniálhatják az átmenetet.

Az általánosítás mellett az *aggregáció* is alkalmazható állapotok esetén, mely „lehetővé teszi az állapot felbontását ortogonális”, azaz egymásmelletti „komponensekre”^[95]. „Az aggregáció egy és-viszony.” „Az aggregált állapot egy állapot az egyik diagramon, és egy másik állapot egy másik diagramon, és egy-egy állapot minden más diagramon.”^[99]



48. ábra: Párhuzamosság

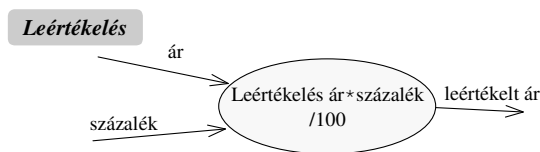
Az állapotok részekre bontásának egyik oka az, hogy egy objektum bizonyos részei párhuzamosan is válthatják részállapotaikat. Maguk a „rendszer objektumai örökletesen konkurensnek”, azaz párhuzamosan működnek, így „az állapotaikat egymástól függetlenül is változtathatják.” „A teljes rendszer állapota nem reprezentálható egyetlen objektum egyetlen állapotával; az az összes objektum állapotainak a szorzata”^[99], amelyen most Descartes (direkt) szorzatot értünk. „Egyetlen objektum állapotán belüli konkurencia megjelenik, ha az objektum felbontható az állapotok és kapcsolatok olyan részalmazaira, hogy mindegyiknek önálló aldiagramja van. Az objektum állapota minden egyes aldiagramon egy állapotot jelent”^[99].

Funkcionális modell

Az OMT utolsó és a gyakorlatban alig használt modellje a funkcionális modell: „a funkcionális modell a rendszeren belüli számításokat írja le”, azt, hogy „az input értékekből az output értékeket hogyan származtatjuk egy számítással”^[123]. „A funkcionális modell tartalmazza az objektummodell értékei közötti megszorításokat is.”^[123]

A funkcionális modell alapeszköze az adatfolyam-diagram, amely az állapot-diagramhoz hasonlóan egy régebbi és elterjedten használt jelölés-rendszer módosítása az objektumorientált koncepcióknak megfelelően. „Az adatfolyam-diagram a rendszer által számított értékek funkcionális viszonyait ábrázolja, beleértve az input- és output-értékeket és a belső adattárolókat.”^[124]

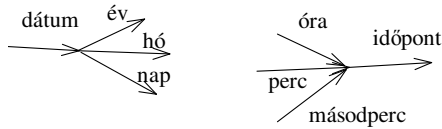
Funkcionális viszonyokon általában számításokat, értéktranszformációkat értenek, melyeket eljárásokként adhatunk meg. Az adatfolyam-diagram alapelemei így az *eljárások*, amelyeket a közöttük áramló *adatokkal* kapcsolunk össze. A diagram határáról indulnak el a bemeneti értékek, illetve a diagram határán túl mutatnak a kimeneti értékek.



49. ábra: Eljárás és adatfolyam

A korábbi technikákhoz hasonlóan, itt is megtalálható az elemek felbonthatósága. Egy adatfolyamdiagramon ábrázolt „teljes adatfolyam egy magas szintű eljárás”^[125], illetve fordítva, „egy eljárás kifejezhető egy újabb adatfolyamdiagramon”^[129].

Az összetett, az ún. „aggregált adatérték” egységként és elemenként is kezelhető, így a jelölésekkel „felbontható komponenseire” illetve fordítva^[126].



50. ábra: Értékfelbontás és egyesítés

A hatások szemszögéből az adatfolyamdiagram egyetlen összetett hatás belső mechanizmusát, részleteinek felbontását, illetve informatikai közegben történő reprezentációját tartalmazza. A diagramon tovább nem bontott hatás-elem az eljárás, az azokat összekapcsoló adatfolyamok pedig a hatás terjedését reprezentálják, amelyeket értékekké (a lehetséges tartomány halmazán belüli indexekké) kvantálva ábrázolunk.

Absztrakció és kompozíció

Az OMT, azaz az Objektum Modellezési Technika, a technikái és eszközei között igen gyakran alkalmazza az absztrakciót, valamint a részekre bontásként, a szóhasználat szerint: aggregátumként történő ábrázolást. Mindkét eszköz a bonyolultság kezelésének az eszköze.

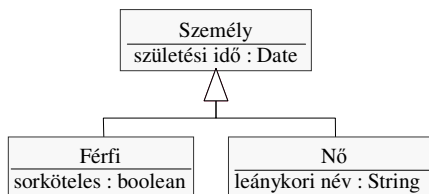
Az összetett rendszereket így alapvetően két módszerrel lehet áttekinthetőbbé egyszerűsíteni.

- Az *absztrakció* (szó szerint: elvonatkoztatás) segítségével eltekinünk a lényegtelennek ítélt részletektől, így a vázlatosabb kép csak a fontosnak tartott elemeket tartalmazza.
- *Részekre bontás* esetén a rendszert lazábban kapcsolódó elemekre osztjuk fel, mely részek (a kapcsolódási módok ismeretében) önállóan is vizsgálhatók.

Gyakran előfordul, hogy bizonyos elemek absztrakt képe megegyezik, mivel az elemek mind egy közös általánosnak a változatai. Ezek az elemek közös jellemzőkkel rendelkeznek, magyarul: *hasonlítanak* egymásra.

Ha a jelölés erre lehetőséget ad, akkor a hasonló elemek ismétlődő közös jellemzői az általános elemhez kiemelhetők. Ezért az absztrakció a *kiemelés* lehetőségét is jelenti, mely felhasználásával ugyanazt a rendszert számotte-

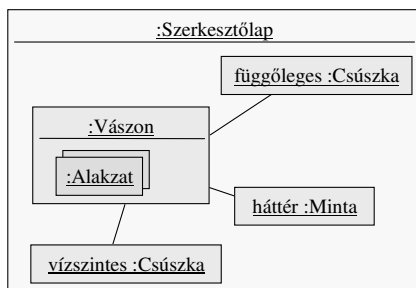
vően rövidebben írhatjuk le. Ugyancsak a kiemeléssel biztosítható a rendszer konzisztenciája, mivel ennek hiányában az általános elemen végrehajtott módosítást minden egyes változaton is végre kellene hajtani.



51. ábra: Absztrakció

Ahogy az „egész” a részeit, úgy az absztrakció is az elemek egy csoportját határozza meg, de

- az absztrakció „vagy”-jellegű csoportosítást jelent, mivel egy absztrakt elem az „vagy” az egyik, „vagy” a másik változat,
- az egész a részeit pedig „és”-jellegű módon csoportosítja: az egész az egyik részből „és” a másik részből áll.

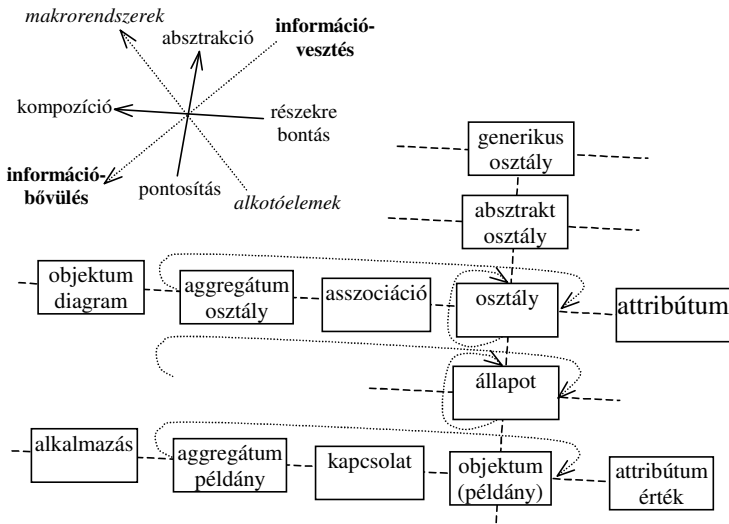


52. Ábra: Egész és részei

Mindkét technika csoportosítást határoz meg, ezért az általánosabb elemet és a több elemből felépülő, összetett „egész” elemet egyaránt *alapelemnek* vagy *makroelemnek*, az általánosnak a változatait, illetve egy összetett elem részeit pedig *alelemeknek* vagy *alkotóelemeknek* nevezzük.

A szemlélettel az OMT strukturális, az objektummodellen ábrázolt elemeinek viszonyai a következő ábrán szemléltethetők. A következők-

ben kompozíció a hétköznapi értelmezéshez közelebb álló, így az UML fogalmánál bővebb, nagyjából az aggregáció jelentésének megfelelő fogalmat értünk.



53. ábra: Az objektummodell foglmainak rendszere.

Az objektum természetesen általában nem az attribútum-értékek kompozíciója, bár ez is egy implementációs lehetőség, illetve az UML a komponenseknek az attribútum-jellegű megadását is engedélyezi. Ebben az esetben az attribútumok által megadott szempontok egyesítéséről van szó, amely a többszörös származtatásnak felel meg. A többszörös származtatás a kompozíciótól csak abban tér el, hogy többszörös származtatás esetén az elemek ugyanarra, kompozíciónál pedig különböző részekre vonatkoznak.

A diagramon jobbról balra az elemek az egyediség figyelembevételével változnak. Az attribútum-értékek csak értékek, így ott nincs egyediség, szemben a valódi objektumokkal. A kapcsolat ugyancsak kompozíció-jellegű, így bizonyos műveletekben a kapcsolódó objektumok közösen vesznek részt, azonban itt nem, csak az aggregációnál jelenik meg az egyediség. A rendszerterv szempontjából a működő teljes alkalmazás esetén sem beszélünk egyediségről, mivel annak univerzum-jellegénél fogva azt nem kell mástól megkülönböztetnünk.



Intelligencia, mint a megértés és az absztrakció képessége

Az intelligencia, illetve a mesterséges intelligencia kérdése az informatika egyik legrégebbi problémája. Mit értünk, illetve mit érthetünk intelligencia alatt? A szabályosságok, azaz az ismétlődő szabályok azonosításának, és azok alkalmazásának fogalmai lehetővé teszik az intelligencia egyfajta megközelítését, melyet munkahipotézisként tekintve elgondolkodhatunk az ily módon értelmezett intelligencia működésén és természetén.

Mesterséges intelligencia

Az elektronikus elven működő számítógépek megjelenésük óta töretlenül bővülő számítási és adattárolási kapacitást tesznek elérhetővé. A mind nagyobb teljesítményű eszközökkel egyre bonyolultabb feladatok fogalmazhatók meg, így indulhattak el és így válhattak a kor nagy „divatjává” azok a kísérletek, melyek egyszerűbb vagy összetettebb számítógépes modelleket alkotva próbáltak meg behatolni a gondolkodás titkaiba. „Mesterséges intelligencia” lett a gyűjtőneve ezeknek a kísérleteknek, melyek eredményeiként sakkozó, tételbizonyító, következtető, matematikai feladatokat megoldó gépek születtek. Az *elnevezés jogosságát* sokan megkérdőjelezik, vagy egyszerűen nagyképűnek tartják (ezért is van idézőjelek közé zárva) arra hivatkozva, hogy ezek a programok éppen olyan megoldásokkal és azokat az eseteket fedik le, melyekhez nem szükséges intelligencia, hanem csak bizonyos utasítások mechanikus végrehajtása. Ezzel azonban pontosan az algoritmusok tulajdonságát írták le. Így, ha a gondolkodás algoritmusként megfogalmazható (algoritmizálható), akkor ott pontosan az adott utasításokat kell mechanikusan végrehajtani. De vajon megoldható-e az algoritmizálás? És az így leírt algoritmus megvalósítható-e számítógéppel? Az eddigi kísérletek az intelli-

gens viselkedés egy-egy részterületen való szimulálására születtek. Így attól függően, hogy milyen elemeket és azokat milyen szinten valósították meg, olyan mértékben és területen adták meg a gondolkodás egy-egy modelljét. De: mindegyik az intelligencia modellje. Egyfelől tehát az elnevezés jogos. Másfelől azonban újabb kétségek merülhetnek fel: egy ilyen egyszerűsített modell tekinthető-e intelligenciának?

A kísérletek egyben tükrözik a tervezők intelligenciáról alkotott elképzeléseinek is, így a *minden* megvalósításban rejlő *közös bázis* valószínűleg az egyik legalapvetőbb eleme az intelligenciának. Általában úgy tartják, hogy ez a legalapvetőbb eszköz a következtetés. A következtetéshez azonban szükséges egy még elemibb funkció, ez pedig az azt megelőző lépésnek, a helyzetfelismerésnek a fogalomköre. Az input adatként reprezentálódó külső világnak és belső állapotoknak azonosítása, „kódolása”, azaz modellként való belső újraépítése az alapja éppen úgy az alakfelismerő automatáknak, mint a tételbizonyító és a szakértői rendszereknek.

Mit is értünk azon, hogy intelligencia?

Különös, hogy az egyik leginkább emberinek tartott tulajdonságnak mind a mai napig nincs pontos definíciója, pedig kizárólag csak a meghatározására több konferenciát is szerveztek.

Az intelligencia mérésével is foglalkozó *pszichometria* megfogalmazása szerint: az intelligencia az, amit az intelligencia-tesztek mérnek. Ez azonban gyakorlatilag nem tekinthető definíciónak, hiszen egy eltérő teszt esetén már mást kellene értenünk az intelligencia fogalma alatt. Mérések esetén a szempont az elsődleges, s annak kellene meghatároznia a mérés módját.

Az intelligenciát általában bizonyos jellegű *problémamegoldó* képességnek tekintik. A kérdés ebben az esetben az, hogy milyen jellegű problémák megoldását hozzuk kapcsolatba az intelligenciával és melyeket nem. Tekinthető intelligensnek az, aki nem tud sakkozni? És aki nem képes áttekinteni egy matematikai levezetést? Vagy aki nem megfelelően viselkedik valamely helyzetben?

Rendkívül különös az a tény, amit a fantasztikus irodalom csak feltételez, a kriminalisztika viszont megtörtént esetek nagy számával bizonyít, hogy a magasabb intelligencia nem feltétlenül jelent etikusabb életet, a sze-

mélyiség jobbításának a szándékát, vagy akár csak nagyobb empátiát. Egy, az átlagnál nagyobb intelligencia éppúgy megjelenhet alkotó, mint romboló formában, s bizonyos területen való megnyilvánulása semmilyen összefüggésben sincs az általános vagy más területen való viselkedéssel.

A problémamegoldás alapján az intelligenciát a *túléléssel* is kapcsolatba hozzák, a túlélést, mint egy darwini értelemben meglehetősen jól körvonalazható és egzaktul mérhető problémaként értelmezve. Valójában azonban a túlélésnek csak a direkt változata jól körvonalazható és egzaktul mérhető. Mind a közvetett, mind az együttes túlélés már olyan problémákat vet fel, melyek csak igen nehezen tekinthetők át, esetleg nem is elemezhetők. Tipikus példák erre, hogy egy agresszívan csak a saját túlélésére törekvő egyed akár csak a saját agresszivitása is elpusztíthat. Másik példa az önfeláldozás, amely az egyén szempontjából épp a túléléssel ellentétes, azonban megmentheti bizonyos egyedek szerveződését.

Valószínű, hogy a nagyon alacsony intelligencia megnehezíti a túlélést. Azonban az is különös, hogy épp bizonyos magas intelligenciájú elemek „élhetetlenek”, azaz kevésbé gazdag és biztonságos életkörnyezetet alakítanak ki önmaguk számára, mint kevésbé magas intelligenciájú társaik.

Láthatjuk, hogy az intelligencia fogalomkörében meglehetősen nagy a bizonytalanság. Szemléletünk, amely az információval, annak ábrázolásával illetve pontosításával, azaz a tanulással kapcsolatos, lehetőséget ad az intelligencia egyfajta megközelítésére is.

Intelligencia

Szemléletünk alapvetően a rendszerek ábrázolásának, illetve a rendszerekre vonatkozó ismeretek finomításának eszközeit tárgyalja. Amennyiben hipotézisként, vagy inkább csak egy lehetséges megközelítési módként elfogadjuk, hogy az „intelligens viselkedés” alapja a helyzetfelismerés, azaz a helyzetre vonatkozó megfelelő ismeret megszerzése, akkor a szemlélet eszközeit felhasználva elgondolkodhatunk az intelligencia természetén. Még egyszer hangsúlyozzuk, hogy nem az intelligencia teljes körű definíciójára törekszünk, hanem mindössze egy *vélemény* továbbgondolására, amely talán majd egyszer segítheti az intelligencia természetének részletes elemzését.

Fogadjuk el hipotézisként a következőket:

Az **intelligencia** a külsőleg megadott szabályok, vagy a változáson belüli ismétlődések, azaz szabályszerűségek azonosításának a képessége.

A fentiek alapján nagyobb, vagy **magasabb intelligenciájúnak** nevezhetjük a megfigyelők közül azt a megfigyelőt, amely a rendszerek adott körére (azaz adott fogalomkörre) vonatkozóan súlyozottan több tényleges szabályszerűséget állapít meg.

Elemezzük az előző megfogalmazás egyes részeit:

- Az intelligenciát viszonylagosan, azaz más megfigyelőkhöz viszonyítva is értelmezhetjük.
- A **magasabb** intelligenciát csak adott rendszerekre, azaz fogalomkörre vonatkoztatjuk, így más fogalomkör esetén más lehet bizonyos intelligenciák viszonya.
- Több szabályszerűséget kell azonosítani. Erre az egyik lehetőség, ha a magasabb intelligenciájú megfigyelő minden olyan szabályszerűséget azonosít, mint bármely más megfigyelő, valamint további szabályszerűségeket is. Az azonosított szabályoknak megadhatjuk egy súlyozott átlagát, mely mérőszám vegyes esetekben is alkalmas az összehasonlításra.
- Fontos, hogy tényleges szabályszerűségek megállapításáról van szó, azaz például a nem-tartományabsztrakció esetén megjelenő hamis hatás azonosítása, a „babona” nem jelent többletet. A tényleges szabályok, illetve általában a szabályok fontosságának meghatározására ugyancsak alkalmazható a súlyozás.

A fenti megfogalmazások valójában definíciók. Azért használtuk a „hipotézis” elnevezést, mivel az intelligencia mind köznapi értelemben, mind a pszichológia szakmai értelmében széleskörűen használt fogalom. „Hipotézisen” munkahipotézist értünk; célunk, hogy segítségével felderíthessük a „valódi” intelligencia bizonyos jellemzőit.

Nézzük tehát, hogy a megfogalmazásból milyen jellemzőket állapíthatunk meg. Fontos, hogy a „magasabb intelligenciát” mindig csak adott területre, adott fogalomkörre vonatkoztatjuk!

Problémamegoldás. Az intelligenciát alapvetően csak a helyzet azonosításának a képességével definiáltuk. Ennek azonban közvetlen kapcsolata van a problémamegoldással. A problémamegoldás egyrészt a helyzetre vonatkozó szabályok, különösen azok feltételrészének az azonosítását jelenti. Jelentheti még a szabályok egymással való kombinálást, azaz a következtetést is. A szabályok együttes értelmezése, alkalmazása azonban még nem feltétlenül nevezhető következtetésnek. A probléma megoldása jelenti továbbá a megtalált megoldási mód alkalmazását is, amelyet azonban már nem sorolunk az intelligenciához; egy megtalált megoldási módot más is megvalósíthat.

Önmagában az intelligencia alapvetően passzív, illetve „aktivitása” mindössze a helyzet felismerésére vonatkozik. A passzivitást abban az értelemben értjük, hogy az intelligencia fogalmát csak a helyzet azonosítására alkalmazzuk, amely még nem jelent a helyzetbe való tényleges beavatkozást, akciót.

Absztrakció, fogalomalkotás. A magasabb intelligencia jobban azonosítja a rendszerben megjelenő ismétlődéseket, szabályszerűségeket, amelyeket kiemelve önálló fogalmakként kezel. A rendszerben való ismétlődések, szabályszerűségek meghatározása és kiemelése valójában az absztrakció művelete.

A magasabb intelligencia jobban azonosít váratlan helyzeteket, mivel arra vagy a korábbi fogalmait tudja alkalmazni, vagy az új helyzet szabályszerűségeiből alkot újabb fogalmakat, s így meg tudja határozni az új helyzet belső összefüggéseit.

A magasabb intelligencia mélyebben hatol be kiterjesztett környezete szemantikájába. A megfigyelőt, mint a rendszer egy entitását tekintve az legfeljebb önmaga és környezete, azaz együttesen: a kiterjesztett környezete változását észlelheti. A változás észlelése általában valamely interfészen („illesztésen”, jelcsatornán) keresztül, jelekként történik. A jeleket, mint a kiterjesztett környezet változásának leképezéseit tekintve a megfigyelő azonosíthatja a leképezés módját (azaz szabályait), valamint az annak forrásául szolgáló változás szabályszerűségeit. Mindez azt jelenti, hogy a szabályszerűségek meghatározásával az adott formában, adott „szintaktikával” megfogalmazott jelek értelmét, valamint annak szabályszerűségeit, azaz a jelek szemantikáját fel tudja építeni, mégpedig a szabályszerűségek (hasonlóság,

kohézió, struktúra, stb.), és ismétlődések azonosítási képességének a mértékében.

Megismerés. Az előzőek alapján a magasabb intelligenciának több ismerete van környezete változásának szabályszerűségeiről, illetve jobban megismerheti azt.

A magasabb intelligencia azonos változást kisebb méretben (például kevesebb szimbólummal) *fogalmazhat meg*, mivel képes az abban megjelenő ismétlődések és szabályosságok kiemelésére.

A magasabb intelligencia azonos helyen több vagy bonyolultabb változást képes ábrázolni, az előző szabály megfordítása alapján.

Intelligensebb ábrázolásmód. Az előzőek alapján meghatározhatjuk az intelligensebb ábrázolásmód fogalmát, mégpedig olyan megadási formaként, amely a hasonlóságok kiemelésével kevésbé redundáns, így alapvetően kisebb méretű formát állít elő.

Intelligens viselkedés. Ha az intelligencia a következtetésen alapulna, nevezhetnénk-e akkor egy olyan programot intelligensnek, amely nem kombinálja a szabályokat, hanem kizárólag alkalmazza azokat? Mégpedig intelligensebbnek tekintünk egy programot, amely több (lényegi) helyzet esetén a megfelelő módon viselkedik. Ugyanígy egy emberi lény is, kizárólag a neveltetése vagy jelleme alapján „megfelelően” viselkedhet egy adott környezetben, anélkül, hogy közben következtetéseket levonására, vagy a szabályok kombinálására kényszerülne.

Az intelligenciára alkotott megközelítésünk szerint intelligensebbnek nevezhetünk egy programot, ha az a helyzetnek megfelelőbben viselkedik, azaz pontosabban meg tudja állapítani bizonyos helyzetek jellegét és arra alkalmazni tudja a szükséges „megfelelő” viselkedés szabályát.

Túlélés és intelligencia. Az intelligenciára alkalmazott meghatározásunk a túléléssel, illetve az alkalmazkodóképességgel is kapcsolatba hozható. A *túlságosan alacsony* intelligencia nem képes azonosítani a túlélést veszélyeztető helyzeteket, vagy azok megoldási módjait, így biztosan állítható, hogy az az evolúció szempontjából káros. A környezethez való alkalmazkodáshoz, azaz a megfelelő szabályok kialakításához szükséges a környezet szabályainak azonosítása.

A túléléshez kell bizonyos fajta aktivitás is. Korábbi megállapításunk szerint azonban az intelligencia alapvetően passzív, azaz csak a helyzet és esetleg az arra alkalmazandó szabály megállapítását jelenti, de nem feltétlenül annak végrehajtását is.

Próbáljunk meg elképzelni egy „túlságosan magas” intelligenciát! Mivel „az” jobban képes azonosítani az ismétlődő fogalmakat, szabályokat, ezért ez a folyamat szinte a teljes működését igénybe veheti, úgy, hogy akcióra gyakorlatilag nem is marad elég erőforrása. A szabályok és a lehetőségek, azaz a lehetséges javítási módok azonosítása miatt folyamatosan környezete tökéletlenségével és tehetetlenségével ütközik, valamint saját korlátaival, amivel még saját burjánzó ötleteit sem tudja a valóságba átültetni. (Erre láthatunk példát a Csodabogár című filmben – eredeti címén: Phenomenon, 1977, rendezte: Jon Turteltaub, író: Gerald Di Pego, főszereplők: John Travolta, Kyra Sedgwick, Forest Whitaker, Robert Duvall).

Egy kis tautológiával azt mondhatjuk, hogy minden felismerés csak annyiban hasznos, amennyiben azt használhatóvá lehet tenni. Az absztrakt és realizálatlan felismerés így sajnos haszontalan és ebben az értelemben felesleges is, mindaddig, amíg valamely véletlen helyzet meg nem teremti a szükségességét.

A túlságosan magas intelligencia éppen ezért evolúciósan káros, és talán pontosan ezért állította be az evolúció a jelenlegi átlagra, a 100-asnak nevezett IQ-ra, a homály jótékony fátylát borítva a világ tökéletlenségére és elfedve a javíthatóság, a világ jobbá tételének sok-sok lehetőségét, hogy figyelmünk elegendő legyen egy-egy apró részlettel foglalkozni.

Jegyezzük meg, hogy a közvetlen evolúciós haszon nem az egyetlen, s nem is a legmegfelelőbb mérőeszköz a hasznosság és éppígy az intelligencia működésének az elbírálására. Mind az egyedek, mind azok szerveződéseik egy összetett és többszintű „célfüggvénnyel” határozzák meg a hasznosságot, ahol a közvetlen túlélés elsődleges szempontjai mellett a belső életminőség, a felfedezések és az alkotás izgalma, a szépség és harmónia átélése kiemelt szerepet kapnak.



Az informatikát, azon belül is az információs rendszerek fejlesztését tekinthetjük egy olyan katalizátornak, amely problémáival, illetve eszköz-készletével segít kialakítani egy tiszta fogalomrendszert az információ ábrázolásáról, illetve az ismeret-szerzésről.

A számítógép így teremti meg azt az alkímiai laboratóriumot, amely a pszichénél összehasonlíthatatlanul egyszerűbb felépítésével, a reprezentáció megtekinthető formájával, a rendkívül primitív tároló és műveleti alapeszközökkel, a megfogalmazások (programok és adatok) nagy bonyolultságának, ugyanakkor tökéletes pontosságának az igényével lehetővé teszi, hogy az információ reprezentációival „vegytiszta” körülmények között szembesüljünk.

Célunk egy olyan elvi megközelítés felvázolása, amely egyszer elvezethet a rendszerfejlesztés egzakt fogalmi kereteinek kialakításához, valamint az általános értelemben vett ismeret-szerzés és reprezentáció legalapvetőbb fogalmainak matematikai precizitású tisztázásához.

