

Hibridinis išvedimo metodas koncepcinei teisinių žinių bazių analizei

L. Paliulionienė (MII)

ĮVADAS

Pranešime nagrinėjamas teisinių žinių bazės koncepcinės analizės uždavinys. Toks uždavinys išskyla, pavyzdžiui, kai reikia nustatyti rengiamo teisinio dokumento variantų skirtumus. Dokumento variantai formalizuotu pavidalu saugomi žinių bazėje. Jų skirtumai gali būti struktūrinio, redakcinio ar koncepcinio pobūdžio. Pranešime pagrindinis dėmesys skirtas koncepcinių skirtumų nustatymui. Projekte NEMESIS [1] tam siūloma naudoti hibridinį išvedimo metodą, kuriame į formalų išvedimą yra įterpiamos specialios skirtumų analizės procedūros.

TEISINIŲ ŽINIŲ VAIZDAVIMAS IR IŠVEDIMAS

Projekte NEMESIS koncepcinio modeliavimo kalba konstruojama freimų logikos (F-logikos) pagrindu. F-logika [2] – tai formalizmas, jungiantis objektinį žinių vaizdavimo būdą su loginiu išvedimo mechanizmu, grindžiamu rezoliucija. F-programą sudaro taisyklės, kurių forma yra *antraštė* ← *kūnas*, kur *antraštė* yra F-molekulė, *kūnas* yra F-molekulių konjunkcija. F-molekulė turi vieną iš šių pavidalų:

- $S_1:S$, kur S žymi apibendrintą sąvoką, S_1 – jos egzempliorių (konkrečią sąvoką),
 - $S_1::S$, kur S_1 žymi sąvokos S specializaciją,
 - $S[K \rightarrow R]$, kur S žymi sąvokos egzempliorių, K – jo komponentą, R – to komponento reikšmę,
 - $S[K \Rightarrow R]$, kur S žymi apibendrintą sąvoką, K – tos sąvokos komponentą, R – to komponento reikšmių kitimo sritį,
 - $S[K^* \rightarrow R]$, kur S žymi apibendrintą sąvoką, K – tos sąvokos komponentą, kurio reikšmę galima naudoti sąvokos egzemplioriuose neišreikštiniu būdu, R – to komponento reikšmę, kuri pagal nutylėjimą perduodama sąvokos egzemplioriams,
 - $P(S_1, \dots, S_n)$, kur P yra predikato vardas, S_1, \dots, S_n – sąvokos.
- Sąvokos vardas yra identifikacinis terminas, kuris gali turėti argumentus. Komponentai, arba atributai, gali būti daugiareikšmiai; tuomet naudojami ženklai $-\gg$, $=\gg$ ir $*-\gg$.

F-logika įgalina modeliuoti nemonotonines žinias. Tuo tarpu monotoninė F-logikos dalis ekvivalentiška predikatų skaičiavimui, t.y. tarp jų egzistuoja abipusis atvaizdavimas.

Rezoliucijos metodas yra teoremų įrodymo būdas, veikiantis sintaksiniame lygyje. F-logikoje naudojamas rezoliucijos metodas, kuriame tikslas yra viena teorema. Jis

tinka, kai reikia atsakyti į konkrečią užklaušą ir gauti konkretizuotus kintamuosius, nurodančius teigiamo atsakymo sąlygas. Tuo tarpu, norint gauti visas konkrečios situacijos išvadas, reikia įrodyti visas teoremas, įrodomas nagrinėjamoje teorijoje. Projekte NEMESIS tam siūloma naudoti teoremų įrodymo būdą, veikiantį semanti- niame lygyje – *tiesioginį išvedimo metodą*, grindžiamą modelių generavimu [3]. Jis pradeda išvedimą nuo aprašančių situaciją faktų ir generuoja tos situacijos modelį. Šį modelį sudaro visi įrodomi faktai. Tiesioginis išvedimo metodas naudojamas ir teisinių žinių bazių koncepcinėje analizėje.

KONCEPCINĖS ANALIZĖS METODAS

Teisinio dokumento variantų skirtumams nustatyti naudojamos dvi stadijos: (1) fak- tų analizė, kurios metu lyginamos sąvokos ir jų egzemplioriai, (2) taisyklių analizė, kurios metu iš vienos taisyklės prielaidų generuojama testuojama situacija (faktų aibė), gaunamos išvados iš šios situacijos, ir jos analizuojamos taip, kaip faktų ana- lizės stadijoje. Taigi, hibridiniame išvedimo metode jungiami tiesioginis išvedimo metodas ir faktų skirtumo analizės procedūra.

Būtina žinių bazių koncepcinės analizės prielaida – žinių bazės turi turėti bendrą branduolį (žodyną, tezaurą), kuriame būtų formalizuotos pagrindinės sąvokos.

Faktų analizėje naudojami požiūrių integravimo principai, kurie taikomi projek- tuojant duomenų bazes ir sudarant globaliąją koncepcinę schemą iš dalinių koncepc- cinių schemų [4]. Dviejų žinių bazių faktai lyginami komponentas po komponento. Pirmiausia lyginamos sąvokos, jų atributai ir egzemplioriai, po to ryšiai tarp sąvokų ar egzempliorių, nusakyti predikatais. Rezultatas yra vektorius, apjungiantis visus lyginimo kriterijus. Pagal šį vektorių nustatomas atstumas tarp sąvokų.

Prieš pradėdant faktų analizę, atliekamas vidinis išvedimas kiekvienoje žinių bazėje, taikant tiesioginį išvedimo metodą, grindžiamą modelių generavimu. Taip pašalinami kai kurie skirtumai, atsiradę dėl to paties dalyko skirtingo formalizavimo.

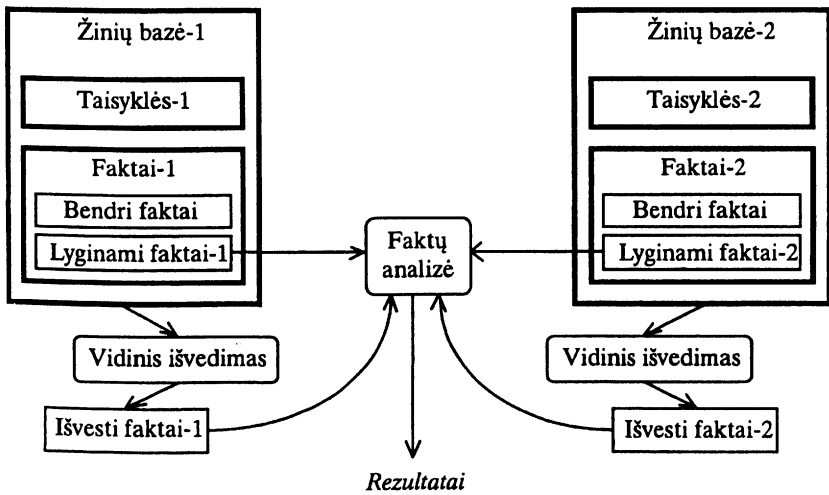
Pavyzdys

1 variantas	a1:a	2 variantas	a1:a
	a2:a		a2:a
	b1(a1):b		b1(X):b ← X:a
	b1(a2):b		

Atlikus vidinį išvedimą antrame variante, abiejų variantų faktų aibės bus vienodos.

Atlikus vidinį išvedimą, išskiriama faktų aibių sankirta, į kurią patenka sintak- siškai vienodi faktai. Lyginimas atliekamas tik tarp likusių faktų aibių (1 pav.). Sąvokų lyginimo žingsniai yra tokie:

1. Nustatoma, kokios sąvokos yra apibrėžtos tik viename variante.
2. Lyginami sąvokų, kurios apibrėžtos abiejuose variantuose, atributų rinkiniai signatūrose. Rezultatas yra nesutampnčių atributų aibės.
3. Lyginamos atributų, kurie apibrėžti abiejuose variantuose, reikšmių kitimo sritys ir priimamos pagal nutylėjimą reikšmės. Galimi šie skirtumai: (a) viename



1 pav.

variante atributas yra vienareikšmis, kitame – daugiareikšmis, (b) reikšmių kitimo sritys susietos apibendrinimo santykiu, t.y. viena reikšmių kitimo sritis yra kitos specializacija arba jos abidvi yra trečios sąvokos specializacijos, (c) reikšmių kitimo sritys nesusietos tarpusavyje, (d) nutylimos reikšmės yra skirtingos, arba viename variante tokia reikšmė apibrėžta, o kitame ne. Rezultatas yra skirtumų aprašas.

Atskirą atvejį sudaro sąvokos, aprašančios straipsnius. Nuoroda į skirtingus straipsnius gali byloti ne apie semantinį, o apie struktūrinį skirtumą. Todėl tikrinamas tokių straipsnių turinys – jei jis sutampa, tai nuorodos laikomos vienodomis.

Sąvokų egzemplioriai lyginami panašiai, tik nagrinėjamos atributų reikšmės, o ne jų kitimo sritys.

Prieš pradėdant *taisyklių analizę*, taip pat išskiriama taisyklių aibių sankirta, į kurią patenka sintaksiškai vienodos taisyklės. Testuojamos situacijos generuojamos tik iš likusių taisyklių.

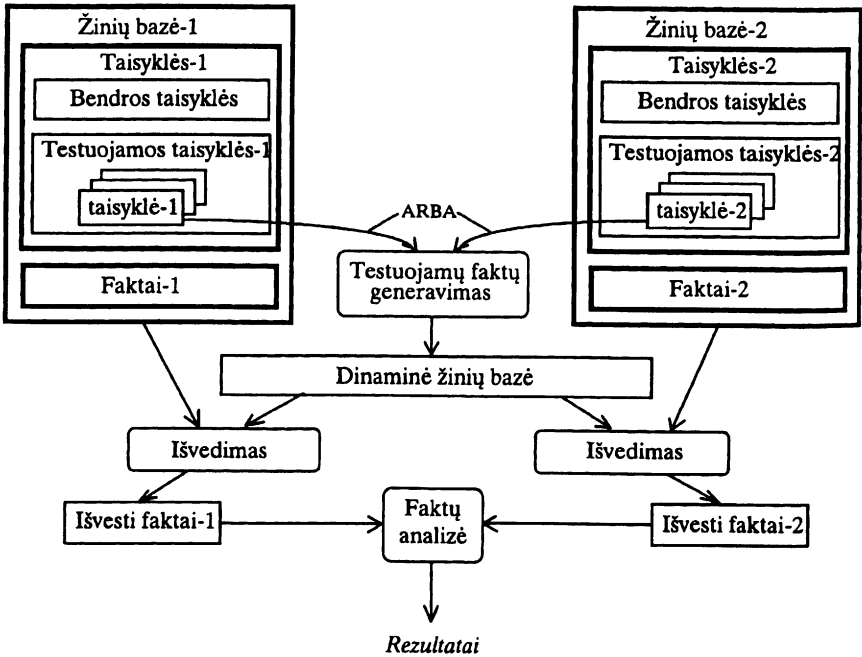
Taisyklių analizės vieną iteraciją sudaro tokie žingsniai (2 pav.):

1. Pasirinktos taisyklės antecedento kintamieji konkretizuojami prototipiniais (bet kaip įvardytais) egzemplioriais iš atitinkamų klasių taip, kad taisyklės prielaidos būtų tenkinamos. Gauti faktai įrašomi į dinaminę žinių bazę. Jei taisyklės antecedente naudojama faktų disjunkcija, tai sudaromi atskiri faktų rinkiniai kiekvienam disjunkcijos elementui.

2. Atliekamas tiesioginis išvedimas iš dinaminės žinių bazės faktų abiejuose variantuose.

3. Faktams–išvadam taikoma faktų analizės procedūra.

Taip viena po kitos apdorojamos abiejų variantų taisyklės, nepriklausančios taisyklių aibių sankirtai.



2 pav.

Pavyzdys. Pademonstruosime siūlomo metodo taikymą Lietuvos Respublikos Baudžiamojo kodekso projekto vieno straipsnio dviejų variantų skirtumams nustatyti:

1 variantas

24 straipsnis. Pasikėsinimas padaryti nusikalstamą veiką

1. Pasikėsinimas padaryti nusikaltimą ar baudžiamąjį nusižengimą yra tyčinis veiksmas, kuriuo tiesiogiai pradedama daryti nusikalstamą veiką, jeigu ji nebuvo baigta dėl aplinkybių, nepriklausančių nuo kaltininko valios.

2. Pasikėsinimas padaryti nusikaltimą yra ir tuomet, kai kaltininkas nesuvokia, kad jis savo veiksmų negali pabaigti todėl, kad kėsinaisi į netinkamą objektą arba naudoja netinkamas priemones.

3. Atsakomybė už pasikėsinimą atsiranda pagal tą baudžiamąjį įstatymą, kuris numato atitinkamą baigtą nusikaltimą, o taip pat šio straipsnio pirmąją dalį. Bausmė už pasikėsinimą gali būti paskirta švelnesnė, negu numatyta už pabaigtą nusikalstamą veiką.

2 variantas.

24 straipsnis. Rengimasis ir pasikėsinimas padaryti nusikalstamą veiką

1. Rengimasis padaryti nusikaltimą yra priemonių ir įrankių suieškojimas ar prietaisų taikymas, veikimo plano sudarymas, bendrininkų suieškojimas arba kitoks tyčinis

sąlygų, palengvinančių nusikaltimo padarymą, sudarymas. Baudžiamoji atsakomybė galima tik už rengimąsi padaryti sunkų ar labai sunkų nusikaltimą.

2. Pasikėsinimas padaryti nusikaltimą ar baudžiamąjį nusižengimą yra tyčinis veiksmas, kuriuo tiesiogiai pradedama daryti nusikalstamą veiką, jeigu ji nebuvo baigta dėl aplinkybių, nepriklausančių nuo kaltininko valios.

3. Pasikėsinimas padaryti nusikaltimą yra ir tuomet, kai kaltininkas nesuvokia, kad jis savo veiksmų negali pabaigti todėl, kad kėsinasi į netinkamą objektą arba naudoja netinkamas priemones.

4. Atsakomybė už rengimąsi ar pasikėsinimą atsiranda pagal tą baudžiamąjį įstatymą, kuris numato atitinkamą baigtą nusikaltimą ar baudžiamąjį nusižengimą, o taip pat šio straipsnio pirmąją arba antrąją dalį.

Pavyzdyje daroma prielaida, kad sąvokos, į kurias daromos nuorodos, yra aprašytos. Predikatų argumentai užrašyti su sematiniais linksniais dėl vaizdumo. Paryškitos sąvokos, formalizuojančios straipsnių ir punktų numerius, vaidina modulių vardų vaidmenį ir turi specialią paskirtį – išvedimo metu jos naudojamos pagrindimo generavimui.

1 varianto formalizavimas:

straipsnis(24,1,bk)

pasikėsinimas::veika[kaltės_forma *→ tyčia, pradeda_ką => nusikalstama_veika
[ar_baigta *→ ne, nebaigimo_priežastis= >>aplinkybė
[apibūdinimas*→nepriklauso_nuo_kaltininko_valios].

straipsnis(24,2,bk)

pasikėsinimas::veika[kaltės_forma *→ tyčia, pradeda_ką => nusikalstama_veika
[ar_baigta *→ ne, nebaigimo_priežastis= >>aplinkybė
[apibūdinimas*→kėsinamasi_į_netinkamą_objektą].
pasikėsinimas::veika[kaltės_forma *→ tyčia, pradeda_ką => nusikalstama_veika
[ar_baigta *→ ne, nebaigimo_priežastis= >>aplinkybė
[apibūdinimas*→naudojamos_netinkamos_priemonės].

straipsnis(24,3,bk)

atsako(kas→N, už_ką→P, pagal_ką→S)
←P:pasikėsinimas[subjektas→N, pradeda_ką→V],
S:baudžiamasis_įstatymas[už_ką→V], V:nusikaltimas.
atsako(kas→N, už_ką→P, pagal_ką→straipsnis(bk,self,1)
←P:pasikėsinimas[subjektas→N].
mažiau_lygu(kas→bausmė(P), už_ką→bausmė(V))
←P:pasikėsinimas[pradeda_ką→V].

2 varianto formalizavimas:

straipsnis(24,1,bk)

rengimasis::veika[kaltės_forma *→ tyčia, rengia_ką=>nusikaltimas,
veiksmas=>pasirengimo_veiksmas].
priemonių_sueškojimas:pasirengimo_veiksmas.

įrankių_sueiškojimas:pasirengimo_veiksmas.
 priemonių_pritaikymas:pasirengimo_veiksmas.
 įrankių_pritaikymas:pasirengimo_veiksmas.
 veikimo_plano_sudarymas:pasirengimo_veiksmas.
 bendrininkų_sueiškojimas:pasirengimo_veiksmas.
 V:pasirengimo_veiksmas
 ← V:veika[tikslas→palengvinti_nusikaltimo_padarymą].
 neatsako(kas→N, už_ką→R, pagal_ką→S:baudžiamasis_įstatymas)
 ← R:rengimasis[subjektas→N, rengia_ką→V],
not V:veika[sunkumas→sunkus],
not V:veika[sunkumas→labai_sunkus].

straipsnis(24,2,bk)

(kaip 1 variante **straipsnis(24,1,bk)**)

straipsnis(24,3,bk)

(kaip 1 variante **straipsnis(24,2,bk)**)

straipsnis(24,4,bk)

atsako(kas→N, už_ką→P, pagal_ką→S)
 ←P:pasikėsinimas[subjektas→N, pradeda_ką→V],
 S:baudžiamasis_įstatymas[už_ką→V],
 V:nusikaltimas **or** V:baudžiamasis_nusižengimas.
 atsako(kas→N, už_ką→R, pagal_ką→S)
 ←R:rengimasis[subjektas→N, rengia_ką→V],
 S:baudžiamasis_įstatymas[už_ką→V].
 atsako(kas→N, už_ką→R, pagal_ką→straipsnis(bk,self,2))
 ←R:rengimasis[subjektas→N].
 atsako(kas→N, už_ką→P, pagal_ką→straipsnis(bk,self,2))
 ←P:pasikėsinimas[subjektas→N].

Faktų analizės metu bus nustatyta, kad antrame variante yra sąvokos, kurių nėra pirmame variante. Tai yra sąvokos „rengimasis“, „pasirengimo_veiksmas“ ir pastarosios sąvokos egzemplioriai.

Kaip veikia taisyklių analizė, panagrinėkime, atlikę iteraciją su pirma taisykle iš straipsnio antro varianto ketvirto punkto. Kadangi taisyklės antecedente naudojama disjunkcija, bus sudaryti du testuojamų faktų rinkiniai:

{p:pasikėsinimas[subjektas→n, pradeda_ką→v],
 s:baudžiamasis_įstatymas[už_ką→v], v:nusikaltimas}
 ir
 {p:pasikėsinimas[subjektas→n, pradeda_ką→v],
 s:baudžiamasis_įstatymas[už_ką→v],
 v:baudžiamasis_nusižengimas}.

Atlikus tiesioginį išvedimą iš pirmo faktų rinkinio, gausime tokias išvadas:

1 variantas: {atsako(kas→n, už_ką→p, pagal_ką→s),
atsako(kas→n, už_ką→p, pagal_ką→straipsnis(bk,self,1),
mažiau_lygu(kas→bausmė(p), už_ką→bausmė(v))}.

2 variantas: {atsako(kas→n, už_ką→p, pagal_ką→s),
atsako(kas→n, už_ką→p, pagal_ką→straipsnis(bk,self,2))}

Atliekant gautų faktų analizę, pirmi abiejų variantų faktai nenagrinėjami, nes jie yra sintaksiškai vienodi. Lyginant antrus faktus, nustatoma, kad atributas “pagal_ką” turi skirtingas reikšmes, bet kadangi tai yra nuorodą į straipsnį, tai išanalizavus straipsnių turinį nustatoma, kad faktai yra semantiškai vienodi. Tarp skirtumų lieka tik trečias pirmo varianto faktas, kuris neturi analogo antrame variante.

Analogiškai daromas išvedimas iš antro testuojamų faktų rinkinio. Atlikę faktų išvadų analizę matome, kad pirmame variante už kėsinimąsi padaryti baudžiamąjį nusižengimą numatyta atsakomybė tik pagal 24 straipsnį, o antrame variante – dar ir pagal straipsnį, numatantį baigtą baudžiamąjį nusižengimą. Įstatymo rengėjas, kuriam bus pateiktas toks analizės rezultatas, greičiausiai padarys išvadą, kad pirmame variante yra spraga.

BAIGIAMOSIOS PASTABOS

Teisinių žinių bazių koncepcinės analizės metodą galima taikyti ne tik variantų skirtumams nustatyti, bet ir su tam tikrais patikslinimais kitais atvejais, kai nagrinėjamos kelios žinių bazės, pavyzdžiui, analizuojant įstatymų suderinamumą arba integruojant įstatymo dalis, parengtas skirtingų autorių:

1. Analizuojant įstatymų suderinamumą, skirtingais laikomi tik faktai, nesuderinami pagal iš anksto apibrėžtą nesuderinamumo santykį. Šis santykis gali būti aprašytas sintaksiškai F-logikos terminais (pavyzdžiui, vienareikšmis atributas negali turėti dvi skirtingas reikšmes) arba semantiškai, tiesiogiai nurodant, kas šioje dalykinėje srityje laikoma nesuderinamu.

2. Integruojant įstatymo dalis, parengtas skirtingų autorių, taip pat skirtingais laikomi tik nesuderinami faktai, o kiti faktai apjungiami. Tai įgalina taikyti lokalizacijos principą, kai sąvokos, atributai ir predikatai aprašomi ten, kur jie naudojami, o vėliau apjungiami globalioje žinių bazėje.

LITERATŪRA

- [1] A. Čaplinskas, L. Paliulionienė, S. Sinkevičiūtė, Project NEMESIS: Knowledge-based aspects in law engineering systems, *DEXA'95 Workshop Proceedings*, Norman Revell, A Min Tjoa (eds.), 1995, p. 413–421.
- [2] M. Kifer, G. Lausen, J. Wu, *Logical Foundations of Object-Oriented and Frame-Based Languages*, Technical Report 93/06, Department of Computer Science, University SUNY at Stony Brook, 1993.
- [3] A. P. Ambrosio, E. Métails, J.-N. Meunier, The linguistic level: Contribution for conceptual design, view integration, reuse and documentation. *Data & Knowledge Engineering*, 1997, 21, p. 111–129.

- [4] L. Paliulionienė, Tiesioginis išvedimo metodas situacijos teisinėms pasekmėms įvertinti teisės inžinerijos sistemose, *Lietuvos mokslas ir pramonė: Informacinės technologijos '97*, Konferencijos pranešimų medžiaga, Kaunas, 1997 (pateikta spaudai).

A hybrid derivation method for the conceptual analysis of legal knowledge bases

L. Paliulionienė (MI)

In knowledge-based legislative engineering, the conceptual analysis of legal knowledge bases is to be used to compare and detect differences of variants of the legal document being prepared, as well as in other cases (to integrate parts of a legal document, to analyse the compatibility of two legal documents, etc.). In project NEMESIS, variants of a legal document are formalised and stored in knowledge bases. The knowledge representation formalism is based on F-logic. To compare the variants, we propose a hybrid derivation method, which integrates formal derivation with difference analysis procedures.

There are two stages of the use of the method: (1) the analysis of facts, when concepts and their instances are compared, (2) the analysis of rules, when consequences of rules are analysed. During one iteration of the analysis of rules, premises of a rule are used to make a test situation (a set of facts), then consequences of the situation are derived by bottom-up theorem proving based on model generation, and the consequences are analysed as facts at the first stage. In this way all rules, that are not common for the variants, are processed.