УДК 004.42:510.69

КОМПОЗИЦІЙНО-НОМІНАТИВНІ ЛОГІКИ З НЕПРЯМИМ ІМЕНУВАННЯМ

Т.В. Россада, О.С. Шкільняк

Київський національний університет імені Тараса Шевченка
01601, Київ, вул. Володимирська, 60
тел.: (044) 259 0511
e-mail: t.rossada@gmail.com, me.oksana@gmail.com

Досліджено композиційно-номінативні логіки, базовані на іменних множинах з непрямим іменуванням. Запропоновано та досліджено операції над такими множинами, зокрема, операції реномінації. На цій основі побудована логіка реномінативного рівня.

In this paper composition-nominative logics based on name sets with indirect nomination are studied. We introduce and investigate operations on such sets, in particular operations of renomination. On this basis we specify a logic of renominative level.

Вступ

Апарат математичної логіки лежить в основі сучасних інформаційних і програмних систем. Такі системи, як правило, базуються на класичній логіці предикатів. Проте класична логіка, незважаючи на її безперечні позитивні якості, має низку обмежень (див., напр., [1]), що ускладнює застосування цієї логіки при розв'язанні нових задач і проблем, що виникають в програмуванні. Тому дуже важливою є проблема розробки нових, програмно-орієнтованих логічних формалізмів. Такими є *композиційно-номінативні* логіки, які будуються на базі спільного для логіки і програмування композиційно-номінативного підходу [2]. На його основі побудовано [1] широкий спектр логічних систем на різних рівнях абстрактності й загальності. Такі рівні відрізняються трактуванням рівня розгляду даних.

Найважливішим типом даних для програмування є *номінати* – дані, побудовані на основі відношення *ім'я* $\mapsto$ *значення*. Імена розглядаються гранично конкретно ("білі" скриньки), значення – абстрактно ("чорні" скриньки). Поняття номінату лежить в основі визначення універсуму $NomD(V,W)$ номінативних даних, який будується індуктивно на основі елементів множини імен V та сукупності (передмножини) базових значень W. У роботі [3] запропонована типологія номінативних даних, яка базується на класифікації фундаментального відношення *ім'я* $\mapsto$ *значення*:

– *значення* класифікуються як прості (неструктуровані) та складні (структуровані),

– *імена* класифікуються як прості (неструктуровані) та складні (структуровані),

– *імена* та значення можуть бути незалежними (пряме іменування), або залежними (можливе непряме іменування).

Зазначені три параметри дають вісім типів номінативних даних. Таку класифікацію названо [3] типологічним кубом номінативних даних. Найпростішими є типи TND_1 (прості імена, прості значення, пряме іменування) та TND_2 (прості імена, прості значення, непряме іменування), найскладнішим – тип TND_8 (складні імена, складні значення, непряме іменування).

Особливо важливим класом номінатів є однозначні – *іменні множини*. Поняття іменної множини (інколи під іншими назвами) активно використовується в математиці й програмуванні. До іменних множин можна віднести [4] кортежі, послідовності, індексовані множини тощо. Традиційно іменними множинами (ІМ) називають множини пар, перша компонента яких – ім'я, а друга – його значення. Формально V-іменну множину над A можна визначити [1] як однозначну функцію вигляду $\delta : V \to A$, де V та A – множини предметних імен та предметних значень. Якщо (в логіці зазвичай так і роблять) вважати $V \cap A = \varnothing$, то такі іменні множини є номінативними даними типу TND_1 (прості імена та значення, пряме іменування). Водночас в програмуванні досить поширеним є непряме іменування, яке часто називають непрямою адресацією.

Непряма адресація є важливою особливістю розробленої К.Л. Ющенко мови адресного програмування, яка була однією з перших мов програмування (див. [5]). Непряма адресація означає, що в комірці пам'яті комп'ютера може знаходитись нова адреса. Наявність операторів та композицій, що опирались на непряму адресацію, дозволило істотно підвищити виразну потужність мови. Адресне програмування дало змогу за допомогою відносно простої мови програмування розробити низку алгоритмів для розв'язку широкого кола задач. Мова адресного програмування успішно використовувалась для побудови інтерпретаторів і трансляторів для перших вітчизняних електронно-обчислювальних машин.

Таким чином, розробка логічних формалізмів, основою яких є іменні множини з непрямим іменуванням, має велике значення для програмування. Саме такі логічні формалізми вивчаються в даній роботі, ідея їх побудови запропонована М.С. Нікітченком. Метою роботи є дослідження іменних множин з непрямим іменуванням та операцій над ними, побудова базованих на них композиційно-номінативних логік реномінативного рівня.

Поняття, які в роботі не визначаються, будемо трактувати в сенсі [1].

1. Іменні множини з непрямим іменуванням

V-іменною множиною з непрямим іменуванням над множиною базових даних A називатимемо однозначну функцію вигляду $\delta : V \rightarrow V \cup A$. Такі множини будемо традиційно подавати у вигляді $[v_1 \mapsto t_1, ..., v_n \mapsto t_n, ...]$, де $v_i \in V$, $t_i \in V \cup A$, $v_i \neq v_j$ при $i \neq j$. Іменні множини з непрямим іменуванням скорочено називатимемо ІМН. Значеннями предметних імен в ІМН можуть бути як предметні імена, так і базові значення.

Клас всіх V-ІМН над A позначаємо ${}^V(V \cup A)$. Клас всіх скінченних V-ІМН над A позначатимемо ${}^V(V \cup A)_F$.

Предикат вигляду $P : {}^V(V \cup A) \rightarrow \{T, F\}$ назвемо V-A-квазіарним предикатом.

Клас V-A-квазіарних предикатів позначимо $Pr^{V,A}$.

Багатократне застосування звичайних ІМ як функцій до певного предметного імені v вже на другому кроці гарантовано дає невизначеність, адже до базових даних ІМ незастосовні. Водночас для ІМН таке застосування означає рух іменною множиною.

Приклад 1. Нехай $d = [x \mapsto y, y \mapsto z, z \mapsto a, v \mapsto b]$, де $x, y, z, v \in V$, $a, b \in A$;
тоді $d(x) = y$, $d(d(x)) = z$, $d(d(d(x))) = a$, $d(d(d(d(x))))\uparrow$.

Для ІМН введемо позначення $d^{(n)}(v)$, де $v \in V$, таким чином:

$d^{(0)}(v) = v$,

$d^{(n+1)}(v) = d(d^{(n)}(v))$, де $n \geq 0$.

Далі, в стилі деномінаційних функцій, $d^{(n)}(v)$ позначатимемо також як ${}^{(n)}v(d)$.

Зокрема, $d(v)$ позначатимемо як ${}^{(1)}v(d)$ та $'v(d)$, $d^{(2)}(v)$ позначатимемо як ${}^{(2)}v(d)$ та $''v(d)$ і т. д.

Шляхом в ІМН d назвемо послідовність $(v_1, v_2, \ldots v_n)$ таку, що $v_i \mapsto v_{i+1} \in d$, $i \in \{1, \ldots, n-1\}$.

Шлях $(v_1, v_2, \ldots v_n)$ позначатимемо також $v_1 \mapsto v_2 \mapsto \ldots \mapsto v_n$.

Циклічний шлях $v_1 \mapsto v_2 \mapsto \ldots \mapsto v_n \mapsto v_1 \mapsto v_2 \mapsto \ldots$ будемо позначати $\{v_1 \mapsto v_2 \mapsto \ldots \mapsto v_n\}$.

Шлях $v_1 \mapsto v_2 \mapsto \ldots \mapsto a$ термінальний, якщо $a \in A$. Шлях $v_1 \mapsto v_2 \mapsto \ldots \mapsto v$ нетермінальний, якщо $v \in V$.

Зокрема, пара $v \mapsto s \in d$ термінальна, якщо $s \in A$, та нетермінальна, якщо $s \in V$.

Ефект циклічного іменування притаманний ІМН.

Приклад 2. Нехай $d = [x \mapsto y, y \mapsto z, z \mapsto x, v \mapsto y, t \mapsto s, u \mapsto w, w \mapsto a]$, де $a \in A$, інші імена $\in V$.

Тоді маємо циклічні шляхи $\{x \mapsto y \mapsto z\}$, $\{y \mapsto z \mapsto x\}$, $\{z \mapsto x \mapsto y\}$.

Окрім того, маємо також шляхи $v \mapsto \{y \mapsto z \mapsto x\}$, $t \mapsto s$, $u \mapsto w \mapsto a$, $w \mapsto a$.

Введемо розгорнуту форму подання ІМН, яку назвемо розгорткою ІМН. Для цього замість кожної пари $v \mapsto t \in d$ записуємо шлях в d, який починається іменем v.

Приклад 3. Наведена в прикладі 2 ІМН $[x \mapsto y, y \mapsto z, z \mapsto x, v \mapsto y, t \mapsto s, u \mapsto w, w \mapsto a]$ має розгортку

$[\{x \mapsto y \mapsto z\}, \{y \mapsto z \mapsto x\}, \{z \mapsto x \mapsto y\}, v \mapsto \{y \mapsto z \mapsto x\}, t \mapsto s, u \mapsto w \mapsto a, w \mapsto a]$.

ІМН *термінально еквівалентні*, якщо їх розгортки мають однакові множини термінальних шляхів.

Надалі приймемо такий природний постулат:

на термінально еквівалентних ІМН предикати приймають однакові значення.

Розгортку ІМН назвемо термінальною, якщо вона містить лише термінальні шляхи.

ІМН, розгортка якої містить лише термінальні шляхи, є мінімальною в множині термінально еквівалентних ІМН у тому розумінні, що не містить "зайвих" компонент, на які не реагують предикати.

Наприклад, серед ІМН, термінально еквівалентних наведеній вище ІМН прикладу 2, такою мінімальною буде ІМН $[u \mapsto w, w \mapsto a]$, розгортка якої – $[u \mapsto w \mapsto a, w \mapsto a]$.

Операції над ІМН. Для ІМН як для бінарних відношень на $V \times (V \cup A)$ можна ввести традиційні операції проекції за координатами:

$pr_1(d) = \{v \in V \mid v \mapsto t \in d$ для деякого $t \in V \cup A\}$;

$pr_2(d) = \{t \in V \cup A \mid v \mapsto t \in d$ для деякого $v \in V\}$.

Можна також ввести традиційні теоретико-множинні операції перетину $\cap$ та різниці \.

Водночас операція об'єднання $\cup$ не завжди є визначеною, адже при об'єднанні ІМН може порушитись умова однозначності.

ІМН d_1 та d_2 назвемо диз'юнктними, якщо $pr_1(d_1) \cap pr_2(d_2) = \varnothing$.

Для диз'юнктних ІМН операція об'єднання $\cup$ вже коректна.

Надалі для операції об'єднання диз'юнктних ІМН вживатимемо також символ +.

Зауважимо, що операції перетину та різниці можна ввести і для розгорток ІМН, при цьому розгортка перетину та перетин розгорток, взагалі кажучи, різні. Те саме має місце і для операції різниці.

Приклад 4. Нехай $d_1 = [x \mapsto y, y \mapsto z, z \mapsto x, v \mapsto y, t \mapsto s, u \mapsto w, w \mapsto a]$, $d_2 = [y \mapsto z, z \mapsto x, v \mapsto y, x \mapsto v, w \mapsto a]$.

Тоді $d_1 \cap d_2 = [y \mapsto z, z \mapsto x, v \mapsto y, w \mapsto a]$.

Розгортка δ_1 ІМН d_1 наведена в прикладі 3. Розгортка ІМН d_2 має вигляд

$\delta_2 = [\{y \mapsto z \mapsto x \mapsto v\}, \{z \mapsto x \mapsto v \mapsto y\}, \{v \mapsto y \mapsto z \mapsto x\}, \{x \mapsto v \mapsto y \mapsto z\}, w \mapsto a]$.

Розгортка ІМН $d_1 \cap d_2$ має вигляд $\delta = [y \mapsto z \mapsto x, z \mapsto x, v \mapsto y \mapsto z \mapsto x, w \mapsto a]$.

Тоді $\delta_1 \cap \delta_2 = [w \mapsto a]$.

Операцію проекції за множиною імен $X \subseteq V$ визначаємо традиційно:

$d \,\|\, X = \{v \mapsto t \in d \mid v \in X\}$.

Замість $d \,\|\, (V \setminus X)$ та $d \,\|\, (V \setminus \{x\})$ будемо скорочено писати $d \,\|\! -X$ та $d \,\|\! -x$.

Операцію накладки для ІМН вводимо так:

$\delta_1 \nabla \delta_2 = (\delta_1 \,\|\, (pr_1(d_2)) + d_2$.

2. Операція реномінації та її властивості

Визначення операції реномінації (перейменування) для випадку ІМН істотно ускладнюється порівняно із визначенням (див. [1]) операції реномінації звичайних ІМ.

Параметризовану за множиною пар імен операцію реномінації $r^{[v_1 \mapsto x_1, \dots, v_n \mapsto x_n]} : {}^V\!A \to {}^V\!A$ визначають так:

$r^{[v_1 \mapsto x_1, \dots, v_n \mapsto x_n]}(d) = [v_1 \mapsto d(x_1), \dots, v_n \mapsto d(x_n)] \cup (d \,\|\, (V \setminus \{v_1, \dots, v_n\}))$.

Це можна записати у вигляді

$r^{[v_1 \mapsto x_1, \dots, v_n \mapsto x_n]}(d) = d \,\|\! -\{v_1, \dots, v_n\} + [v_1 \mapsto d(x_1), \dots, v_n \mapsto d(x_n)]$.

Замість $r^{[v_1 \mapsto x_1, \dots, v_n \mapsto x_n]}$ зазвичай вживають дещо коротше позначення $r^{v_1, \dots, v_n}_{x_1, \dots, x_n}$.

Замість $y_1, \dots, y_n$ також скорочено пишуть $\overline{y}$. Тоді $r^{[v_1 \mapsto x_1, \dots, v_n \mapsto x_n]}$ можна скорочено позначати як $r^{[\overline{v} \mapsto \overline{x}]}$.

Операція реномінації ІМН мусить враховувати той факт, що імена, які набувають нові значення (ліві компоненти пар імен) та самі ці значення (задаються правими компонентами пар імен) можуть задаватися опосередковано, за допомогою ІМН, до якої застосовується реномінація.

В загальному вигляді операція реномінації ІМН позначається так: $r^{[{}^{m_1}v_1 \mapsto {}^{n_1}x_1, \dots, {}^{m_k}v_k \mapsto {}^{n_k}x_k]}$.

Тут позначення ${}^{n}z$ означає n-кратне застосування вхідної ІМН d (аргумент операції реномінації) до предметного імені z. Зокрема, у випадку $n = 0$ маємо саме ім'я z, у випадку $n = 1$ будемо також писати $'z$, у випадку $n = 2$ писатимемо $''z$ і т.д.

Зауважимо, що в таких позначеннях операція реномінації звичайних ІМ $r^{[v_1 \mapsto x_1, \dots, v_n \mapsto x_n]}$ подаватиметься у вигляді $r^{[v_1 \mapsto 'x_1, \dots, v_n \mapsto 'x_n]}$.

Дамо визначення операції реномінації ІМН $r^{[{}^{m_1}v_1 \mapsto {}^{n_1}x_1, \dots, {}^{m_k}v_k \mapsto {}^{n_k}x_k]} : {}^V(V \cup A) \to {}^V(V \cup A)$.

ІМН $r^{[{}^{m_1}v_1 \mapsto {}^{n_1}x_1, \dots, {}^{m_k}v_k \mapsto {}^{n_k}x_k]}(d)$, яка є результатом застосування операції $r^{[{}^{m_1}v_1 \mapsto {}^{n_1}x_1, \dots, {}^{m_k}v_k \mapsto {}^{n_k}x_k]}$ до ІМН d, задамо таким чином.

Для кожного v_i, $i \in \{1, .., k\}$, обчислюємо $d^{(m_i)}(v_i)$. При цьому можливі три випадки: отримуємо деяке $z_i \in V$, отримуємо деяке $b \in A$, або маємо $d^{(m_i)}(v_i)\uparrow$, тобто значення $d^{(m_i)}(v_i)$ невизначене. Останні два випадки непродуктивні. Формуємо послідовність $q = (z_1, \dots, z_l)$ так отриманих предметних імен, тут z_j відповідає v_{i_j}, де $1 \le i_1 \le \dots \le i_l \le k$. Зрозуміло, що в цій послідовності можливі повтори, це відповідає ситуації $d^{(m_i)}(v_i) = d^{(m_j)}(v_j) \in V$ для різних $i, j \in \{1, .., k\}$. Нехай Z – множина всіх імен послідовності q (зрозуміло, що повтори в множині неможливі).

Для кожного z_j обчислюємо відповідне $d^{(n_{i_j})}(x_{i_j})$, $j \in \{1, .., l\}$. Якщо $d^{(n_{i_j})}(x_{i_j})\uparrow$, то вилучаємо відповідне z_j з послідовності q. Отримуємо послідовність q', в якій залишаються лише імена z_j із обчисленими значеннями $s_j = d^{(n_{i_j})}(x_{i_j})$. Далі формуємо множину пар $z_j \mapsto s_j$ для всіх таких $z_j \in q'$. Перші компоненти в цих парах можуть повторюватися, водночас ІМН – однозначні. Тому треба залишити в такій множині лише такі $z_j \mapsto s_j$, для яких у цій множині немає пари $z_p \mapsto s_p$ із $z_j = z_p$ та $s_j \ne s_p$. Інакше кажучи, із цієї множини пар вилучаємо всі пари з однаковими першими та різними другими компонентами. В результаті отримуємо іменну множину таких пар $\eta = [z_{j_1} \mapsto s_{j_1}, \dots, z_{j_t} \mapsto s_{j_t}]$. Тепер остаточно маємо $r^{[{}^{m_1}v_1 \mapsto {}^{n_1}x_1, \dots, {}^{m_k}v_k \mapsto {}^{n_k}x_k]}(d) = d \,\|\! -Z + \eta$.

Операцію реномінації $r^{[{}^{m_1}v_1 \mapsto {}^{n_1}x_1, \dots, {}^{m_k}v_k \mapsto {}^{n_k}x_k]}$ скорочено позначатимемо $r^{[\alpha_1 \mapsto \beta_1, \dots, \alpha_k \mapsto \beta_k]}$, де α_i та β_i позначають вирази для обчислення непрямих імен та їх значень. Зовсім стисло позначаємо реномінацію як $r^{[\overline{\alpha} \mapsto \overline{\beta}]}$.

Приклад 5. Обчислимо $\delta = r^{['x \mapsto ''''y,\ ''x \mapsto z,\ 'y \mapsto ''z,\ u \mapsto 'y,\ 'u \mapsto 'z]}(d)$, де $d = [x \mapsto p, y \mapsto p, z \mapsto u, u \mapsto b, p \mapsto v, v \mapsto a]$, $a, b \in A$ та $x, y, p, z, u, v \in V$. Маємо $'x(d) = d(x) = p$, $''''y(d) = d^{(4)}(y)\uparrow$, $''x(d) = d(d(x)) = v$, $'y(d) = d(y) = p$, $''z(d) = d(d(z)) = b$, $'u(d) = d(u) = b$, $'z(d) = d(z) = u$. Тоді послідовність продуктивних імен $q = (p, v, p, u)$, множина таких імен $Z = \{p, v, u\}$. Для p маємо два входження в послідовність, проте для першого входження його значення $''''y(d)\uparrow$. Таким чином, отримуємо ІМН пар $\eta = [v \mapsto z, p \mapsto b, u \mapsto p]$. Звідси результуюча ІМН $\delta = d \,\|\! -Z + \eta$ має вигляд: $\delta = [x \mapsto p, y \mapsto p, z \mapsto u, v \mapsto z, p \mapsto b, u \mapsto p]$.

Властивості операції реномінації. Операція реномінації ІМ $\mathrm{r}^{[\bar{v}\mapsto\bar{x}]}$ монотонна за розширенням даних: для всіх $d_1, d_2 \in {}^V\!A$ маємо $d_1 \subseteq d_2 \Rightarrow \mathrm{r}^{[\bar{v}\mapsto\bar{x}]}(d_1) \subseteq \mathrm{r}^{[\bar{v}\mapsto\bar{x}]}(d_2)$. Це вже не так для операції реномінації ІМН $\mathrm{r}^{[\bar{\alpha}\mapsto\bar{\beta}]}$.

Приклад 6. Нехай $\rho = \mathrm{r}^{['x\mapsto'u,\ 'y\mapsto'v]}$. Візьмемо ІМН $d_1 = [x\mapsto z, u\mapsto a, v\mapsto b]$ та $d_2 = [x\mapsto z, y\mapsto z, u\mapsto a, v\mapsto b]$, де $a, b \in A$ та $x, y, z, u, v \in V$. Маємо $'x(d_1) = z$, $'y(d_1)\uparrow$, $'u(d_1) = a$, $'v(d_1) = b$; $'x(d_2) = z$, $'y(d_2) = z$, $'u(d_2) = a$, $'v(d_2) = b$. Тоді $\rho(d_1) = d_1 \,\|{-z} + [z\mapsto a] = [x\mapsto z, u\mapsto a, v\mapsto b, z\mapsto a]$, водночас $\rho(d_2) = d_2 \,\|{-z} + \varnothing = [x\mapsto z, u\mapsto a, v\mapsto b]$. Таким чином, $d_1 \subset d_2$ та $\rho(d_1) \supset \rho(d_2)$ – монотонності для ρ немає.

Візьмемо тепер $d_3 = [x\mapsto z, u\mapsto a, v\mapsto b, z\mapsto b]$, $d_4 = [x\mapsto z, y\mapsto z, u\mapsto a, v\mapsto b, z\mapsto b]$. Тоді $d_3 \subset d_4$. Тепер маємо $\rho(d_3) = d_3 \,\|{-z} + [z\mapsto a] = [x\mapsto z, u\mapsto a, v\mapsto b, z\mapsto a]$, $\rho(d_4) = d_4 \,\|{-z} + \varnothing = [x\mapsto z, y\mapsto z, u\mapsto a, v\mapsto b]$, тому знову монотонності немає, адже невірно $\rho(d_3) \subseteq \rho(d_4)$.

Таким чином:

Твердження 1. Операція реномінації ІМН $\mathrm{r}^{[\bar{\alpha}\mapsto\bar{\beta}]}$ не є монотонною.

На відміну від операцій реномінації ІМ, для операцій реномінації ІМН вже не можна робити їх згортку, тобто однотипно, незалежно від вхідної ІМН, замінювати послідовне застосування двох і більше реномінацій на застосування однієї – їх згортки. Для реномінації ІМ $\mathrm{r}^{[\bar{v}\mapsto\bar{x}]}$ та $\mathrm{r}^{[\bar{u}\mapsto\bar{y}]}$ їх згортка $\mathrm{r}^{[\bar{v}\mapsto\bar{x}]\circ[\bar{u}\mapsto\bar{y}]}$ залежить лише від пар імен початкових реномінацій. Проте неявне, непряме задання імен в реномінаціях ІМН веде до залежності реномінації – результату послідовного застосування кількох реномінацій – не тільки від них, а й від конкретної ІМН, до якої ведеться застосування цих реномінацій. Це засвідчує наступний

Приклад 7. Нехай $d_1 = [y\mapsto u, u\mapsto b, z\mapsto a]$, $d_2 = [y\mapsto v, v\mapsto b, z\mapsto a]$, де $a, b \in A$ та $x, y, z, u, v \in V$.

Розглянемо послідовне застосування реномінацій $\mathrm{r}^{[u\mapsto'z]}$ та $\mathrm{r}^{[x\mapsto''y]}$ до d_1 та до d_2.

Маємо $\mathrm{r}^{[x\mapsto''y]}(\mathrm{r}^{[u\mapsto'z]}(d_1)) = \mathrm{r}^{[x\mapsto''y]}([y\mapsto u, u\mapsto a, z\mapsto a] = [y\mapsto u, u\mapsto a, z\mapsto a, x\mapsto a] = \mathrm{r}^{[u\mapsto'z,\ x\mapsto'z]}(d_1)$;

$\mathrm{r}^{[x\mapsto''y]}(\mathrm{r}^{[u\mapsto'z]}(d_2)) = \mathrm{r}^{[x\mapsto''y]}([y\mapsto v, v\mapsto b, z\mapsto a, u\mapsto a]) = [y\mapsto v, v\mapsto b, z\mapsto a, u\mapsto a, x\mapsto b] = \mathrm{r}^{[u\mapsto'z,\ x\mapsto''y]}(d_2)$.

У випадку d_1 згортка $\mathrm{r}^{[u\mapsto'z]}$ та $\mathrm{r}^{[x\mapsto''y]}$ подається як $\mathrm{r}^{[u\mapsto'z,\ x\mapsto'z]}$, у випадку d_2 – як $\mathrm{r}^{[u\mapsto'z,\ x\mapsto''y]}$.

3. Реномінативна логіка з непрямим іменуванням

Семантичними моделями композиційно-номінативних логік реномінативного рівня є [1] композиційні системи квазіарних предикатів реномінативного рівня, де множина композицій визначається базовими композиціями пропозиційного рівня та композицією реномінації. Досліджувались [1] традиційні реномінативні логіки, в яких композиція реномінації визначається через відповідну операцію реномінації ІМ. У даній роботі пропонуються реномінативні логіки, в яких композиція реномінації $\boldsymbol{R}^{[\bar{\alpha}\mapsto\bar{\beta}]}$ визначається операцією реномінації $\mathrm{r}^{[\bar{\alpha}\mapsto\bar{\beta}]}$ іменних множин з непрямим іменуванням. Такі логіки назвемо *реномінативними логіками з непрямим іменуванням* (РНЛН). Семантичними моделями цих логік є композиційні системи вигляду $({}^V(V\cup A), Pr^{V,A}, \boldsymbol{C})$, де $\boldsymbol{C}$ визначається множиною базових композицій $\neg$, $\vee$ та $\boldsymbol{R}^{[\bar{\alpha}\mapsto\bar{\beta}]}$.

1-арна параметрична реномінації $\boldsymbol{R}^{[\bar{\alpha}\mapsto\bar{\beta}]} : Pr^{V,A} \to Pr^{V,A}$ кожному предикату зіставляє *V-A*-квазіарний предикат $\boldsymbol{R}^{[\bar{\alpha}\mapsto\bar{\beta}]}(P)$, значення якого для кожного $d \in {}^V(V\cup A)$ задається так:

$$\boldsymbol{R}^{[\bar{\alpha}\mapsto\bar{\beta}]}(P)(d) = P(\mathrm{r}^{[\bar{\alpha}\mapsto\bar{\beta}]}(d)).$$

Твердження 2. Композиція $\boldsymbol{R}^{[\bar{\alpha}\mapsto\bar{\beta}]}$ не зберігає еквітонність (монотонність) *V-A*-квазіарних предикатів.

Задамо *V-A*-квазіарний предикат P наступним чином: $P(d) = T$, якщо $z \in pr_1(d)$, та $P(d)\uparrow$, якщо $z \notin pr_1(d)$. Такий предикат P – еквітонний. Розглянемо тепер предикат $\boldsymbol{R}^{['x\mapsto'u,\ 'y\mapsto'v]}(P)$. Цей предикат нееквітонний.

Справді, візьмемо ІМН $d_3 = [x\mapsto z, u\mapsto a, v\mapsto b, z\mapsto b]$ та $d_4 = [x\mapsto z, y\mapsto z, u\mapsto a, v\mapsto b, z\mapsto b]$ прикладу 5. Тоді:

$$\boldsymbol{R}^{['x\mapsto'u,\ 'y\mapsto'v]}(P)(d_3) = P(\mathrm{r}^{['x\mapsto'u,\ 'y\mapsto'v]}(d_3)) = P([x\mapsto z, u\mapsto a, v\mapsto b, z\mapsto a]) = T,$$

$$\boldsymbol{R}^{['x\mapsto'u,\ 'y\mapsto'v]}(P)(d_4) = P(\mathrm{r}^{['x\mapsto'u,\ 'y\mapsto'v]}(d_4)) = P([x\mapsto z, y\mapsto z, u\mapsto a, v\mapsto b])\uparrow.$$

Маємо $d_3 \subset d_4$, $\boldsymbol{R}^{['x\mapsto'u,\ 'y\mapsto'v]}(P)(d_3) = T$ та $\boldsymbol{R}^{['x\mapsto'u,\ 'y\mapsto'v]}(P)(d_4)\uparrow$. Отже, предикат $\boldsymbol{R}^{['x\mapsto'u,\ 'y\mapsto'v]}(P)$ нееквітонний.

Наведемо основні властивості композицій $\boldsymbol{R}^{[\bar{\alpha}\mapsto\bar{\beta}]}$.

R¬) $\boldsymbol{R}^{[\bar{\alpha}\mapsto\bar{\beta}]}(\neg P) = \neg\boldsymbol{R}^{[\bar{\alpha}\mapsto\bar{\beta}]}(P)$;

R∨) $\boldsymbol{R}^{[\bar{\alpha}\mapsto\bar{\beta}]}(P \vee Q) = \boldsymbol{R}^{[\bar{\alpha}\mapsto\bar{\beta}]}(P) \vee \boldsymbol{R}^{[\bar{\alpha}\mapsto\bar{\beta}]}(Q)$.

Аналогічно можна записати властивості R→, R&, R↔ для похідних логічних зв'язок **→**, **&**, **↔**.

RR) $\boldsymbol{R}^{[\bar{\alpha}\mapsto\bar{\beta}]}(\boldsymbol{R}^{[\bar{\gamma}\mapsto\bar{\eta}]}(P))(d) = P(\mathrm{r}^{[\bar{\gamma}\mapsto\bar{\eta}]}(\mathrm{r}^{[\bar{\alpha}\mapsto\bar{\beta}]}(d)))$ для кожного $d \in {}^V(V\cup A)$.

На відміну від традиційних РНЛ предикатів над ІМ, тут властивість RR записується в слабкій формі. Неможливість виконувати згортку композицій реномінації *V-A*-квазіарних предикатів випливає з неможливості виконувати згортку операцій реномінації для ІМН.

Специфічний вигляд має також властивість згортки тотожного перейменування RT.

RT) $\boldsymbol{R}^{[{}^{(n)}x\mapsto{}^{(n+1)}x,\ \bar{\alpha}\mapsto\bar{\beta}]}(P) = \boldsymbol{R}^{[\bar{\alpha}\mapsto\bar{\beta}]}(P)$.

Предикатна композиційна система $({}^V(V\cup A), Pr^{V,A}, \boldsymbol{C})$ задає дві алгебри: алгебру даних $(A, Pr^{V,A})$ та алгебру предикатів $(Pr^{V,A}, \boldsymbol{C})$, терми якої трактуються як формули мови РНЛН.

Мова РНЛН. Опишемо мову РНЛН. Алфавіт мови: множина V предметних імен, множина Ps предикатних символів; символи базових композицій $\neg$, $\vee$, $R^{[\bar{\alpha}\mapsto\bar{\beta}]}$.

Множина Fr формул мови визначається індуктивно:

1) кожний $p\in Ps$ є формулою; такі формули назвемо атомарними;

2) нехай Φ та Ψ – формули; тоді $\neg\Phi$ та $\vee\Phi\Psi$ – формула;

3) нехай Φ – формула; тоді $R^{[\bar{\alpha}\mapsto\bar{\beta}]}\Phi$ – формула.

Задамо відображення інтерпретації формул $J : Fr\to Pr^{V,A}$ визначається за допомогою тотального однозначного відображення $I : Ps\to Pr^{V,A}$, яке виділяє базові предикати, позначаючи їх символами Ps.

1) $J(p) = I(p)$ для кожного $p\in Ps$;

2) $J(\neg\Phi) = \neg(J(\Phi))$, $J(\vee\Phi\Psi) = \vee(J(\Phi), J(\Psi))$;

3) $J(R^{[\bar{\alpha}\mapsto\bar{\beta}]}\Phi) = \boldsymbol{R}^{[\bar{\alpha}\mapsto\bar{\beta}]}(J(\Phi))$.

Об'єкти вигляду $((A, Pr^{V,A}), I)$ називають [1] алгебраїчними системами (АС) з доданою сигнатурою. Відображення $I : Ps\to Pr^{V,A}$ прив'язує алгебру даних $(A, Pr^{V,A})$ до мови РНЛН, тому така АС визначає композиційну систему $({}^V(V\cup A), Pr^{V,A}, \boldsymbol{C})$. Таким чином, АС з доданою сигнатурою є інтегрованими семантичними моделями, які пов'язують мову логіки з алгебрами даних. Такі АС скорочено позначають $\boldsymbol{A} = (A, I)$.

Традиційно позначаємо Φ_A предикат $J(\Phi)$ – значення формули Φ при інтерпретації на моделі мови $\boldsymbol{A}$.

Стандартним чином (див. [1]) вводимо поняття (частково) істинної при інтерпретації на моделі мови $\boldsymbol{A}$ та виконуваної при інтерпретації на $\boldsymbol{A}$ формули, всюди істинної та виконуваної формули.

Формула Φ частково істинна при інтерпретації на $\boldsymbol{A} = (A, \boldsymbol{I})$ (позначаємо $\boldsymbol{A}\models\Phi$) якщо предикат Φ_A – частково істинний. Формула Φ всюди істинна (позначаємо $\models\Phi$), якщо $\boldsymbol{A}\models\Phi$ для кожної моделі мови $\boldsymbol{A}$.

Семантичні властивості РНЛН. На множині формул мови РНЛН стандартним чином (див. [1]) вводяться відношення логічного наслідку $\models$, слабкого логічного наслідку $\Vert\!=$, логічної еквівалентності $\sim$; визначаються поняття тавтології, тавтологічного наслідку. Відношення логічного наслідку поширюється на множини формул.

Для РНЛН справджуються відомі [1] теореми еквівалентності та заміни еквівалентних.

Теорема 1 (еквівалентності). Нехай формула Φ' отримана з формули Φ заміною деяких входжень формул $\Phi_1, ..., \Phi_n$ на формули $\Psi_1, ..., \Psi_n$ відповідно. Якщо $\Phi_1\sim\Psi_1, ..., \Phi_n\sim\Psi_n$, то $\Phi\sim\Phi'$.

Теорема 2 (заміни еквівалентних). Нехай $\Phi\sim\Psi$. Тоді: $\Phi,\Gamma\models\Delta \Leftrightarrow \Psi,\Gamma\models\Delta$ та $\Gamma\models\Delta,\Phi \Leftrightarrow \Gamma\models\Delta,\Psi$.

Семантичні властивості пропозиційного рівня аналогічні відповідним властивостям традиційної реномінативної логіки та класичної пропозиційної логіки (див. [1]). Наведемо властивості, пов'язані з композицією реномінації. Вони записуються згідно відповідних властивостей композиції реномінації *V-A*-квазіарних предикатів над ІМН і мають таку ж назву.

R¬) $R^{[\bar{\alpha}\mapsto\bar{\beta}]}(\neg\Phi) \sim \neg R^{[\bar{\alpha}\mapsto\bar{\beta}]}(\Phi)$.

R∨) $R^{[\bar{\alpha}\mapsto\bar{\beta}]}(\Phi\vee\Psi) \sim R^{[\bar{\alpha}\mapsto\bar{\beta}]}(\Phi)\vee R^{[\bar{\alpha}\mapsto\bar{\beta}]}(\Psi)$.

Узагальнюючи R¬ та R∨, отримуємо властивості RR¬ та RR∨.

RR¬) $R^{[\bar{\alpha}\mapsto\bar{\beta}]}(...R^{[\bar{\gamma}\mapsto\bar{\eta}]}(\neg\Phi)...) \sim \neg R^{[\bar{\alpha}\mapsto\bar{\beta}]}(...R^{[\bar{\gamma}\mapsto\bar{\eta}]}(\Phi)...)$.

RR∨) $R^{[\bar{\alpha}\mapsto\bar{\beta}]}(...R^{[\bar{\gamma}\mapsto\bar{\eta}]}(\Phi\vee\Psi)...) \sim R^{[\bar{\alpha}\mapsto\bar{\beta}]}(...R^{[\bar{\gamma}\mapsto\bar{\eta}]}(\Phi)...)\vee R^{[\bar{\alpha}\mapsto\bar{\beta}]}(...R^{[\bar{\gamma}\mapsto\bar{\eta}]}(\Psi)...)$.

RT) $R^{[{}^{(n)}x\mapsto{}^{(n+1)}x,\ \bar{\alpha}\mapsto\bar{\beta}]}(\Phi) \sim R^{[\bar{\alpha}\mapsto\bar{\beta}]}(\Phi)$.

Записати властивість RR в аналогічному вигляді, із використанням $\sim$, вже неможливо. Можна лише переписати властивість RR композиції реномінації:

RR_C) для кожних $\boldsymbol{A} = (A, \boldsymbol{I})$ та $d\in{}^V(V\cup A)$ маємо $R^{[\bar{\alpha}\mapsto\bar{\beta}]}(R^{[\bar{\gamma}\mapsto\bar{\eta}]}(\Phi_A)(d) = \Phi_A(\mathrm{r}^{[\bar{\gamma}\mapsto\bar{\eta}]}(\mathrm{r}^{[\bar{\alpha}\mapsto\bar{\beta}]}(d))$.

Кожну формулу мови РНЛН можна звести до класичноподібної нормальної форми, коли реномінації просунуті максимально вглиб формули.

Формулу вигляду $R^{[\bar{\alpha}\mapsto\bar{\beta}]}(...R^{[\bar{\gamma}\mapsto\bar{\eta}]}(p)...)$, де $p\in Ps$ та в реномінаціях згорнуті тотожні перейменування, назвемо примітивною.

Формулу мови РНЛН назвемо *нормальною*, якщо вона утворена із примітивних формул за допомогою символів пропозиційних композицій.

Про нормальну формулу кажуть, що вона знаходиться в нормальній формі.

Подібні нормальні форми введені [1] для формул традиційної реномінативної логіки.

Теорема 3. Для кожної формули Φ можна збудувати нормальну формулу Ψ таку, що $\Phi\sim\Psi$.

Використовуючи R¬ та R∨, просуваємо реномінації вглиб формули. Використовуючи RT, згортаємо тотожні перейменування. Кожний такий крок веде до формули, логічно еквівалентної попередній. Згідно теореми еквівалентності, отримана таким перетвореннями нормальна формула еквівалентна початковій.

Існує алгоритм, який дозволяє для кожної формули Φ мови РНЛН з'ясувати, чи є Φ всюди істинною.

Без обмежень загальності, згідно теореми 3, можна розглядати лише нормальні форми.

Нехай нормальна формула Ф така, що $|\neq \Phi$. Тоді для Ф існує контрмодель ($\boldsymbol{A}$, d) – така модель мови $\boldsymbol{A} = (A, \boldsymbol{I})$ та ІМН $d \in {}^{V}(V \cup A)$, що $\Phi_A(d) = F$. Нехай $nm(\Phi)$ – множина всіх предметних імен, які фігурують в символах реномінації формули Ф. Такі імена назвемо явними. Враховуючи непряме іменування, при обчисленні значень предикатів, заданих примітивними формулами, можуть задіюватися неявні для Ф імена, вони фігурують в d. Їх кількість залежить від кількості застосувань d до предметних імен, що описується параметрами реномінації, але для конкретної Ф вона завжди скінченна. Таким чином, для кожної контрмоделі ($\boldsymbol{A}$, d) в ІМН d задіюється не більше певної кількості m елементів (пар ім'я $\mapsto$ значення), таке m залежить від Ф. Тому для пошуку контрмоделі розглядаємо лише скінченну кількість ІМН, які мають не більше m елементів. Компонентами пар будуть імена $nm(\Phi)$ та неявні імена, таких неявних імен не більше $2m$. Для термінальних пар друга компонента (значення) є дублером першої (імені). Для кожного такого d розглядаємо всеможливі значення, які можуть приймати на ньому елементарні підформули Ф (можлива і невизначеність). Для кожного такого набору значень знаходимо (згідно законів пропозиційної логіки часткових предикатів) значення Φ_A. Якщо хоч один раз для Φ_A отримаємо значення F, то $|\neq \Phi$, в іншому випадку $|= \Phi$.

Таким чином, отримуємо:

Твердження 3. Множина всюди істинних формул РНЛН алгоритмічно розв'язна відносно множини всіх формул мови.

Аналогічне твердження про розв'язність множини всюди істинних формул справджується [1] і для традиційної реномінативної логіки.

Висновки

У роботі вивчаються композиційно-номінативні логіки, базовані на іменних множинах з непрямим іменуванням. Запропоновано та досліджено операції над такими множинами, особливу увагу приділено операції реномінації. На цій основі побудована реномінативна логіка з непрямим іменуванням. Дослідження планується продовжити в напрямках побудови числень реномінативних логік та побудови першопорядкових логік з непрямим іменуванням.

1. *Никитченко Н.С.* Композиционно-номинативный подход к уточнению понятия программы // Проблеми програмування. – 1999. – № 1. – С. 16–31.
2. *Нікітченко М.С., Шкільняк С.С.* Математична логіка та теорія алгоритмів. – К.: ВПЦ Київський університет, 2008. – 528 с.
3. *Никитченко Н.С.* Композиционно-номинативные аспекты адресного программирования // Кибернетика и системный анализ. – 2009. – № 6. – С. 24–35.
4. *Басараб И.А., Никитченко Н.С., Редько В.Н.* Композиционные базы данных. – К.: Либідь, 1992. –192 с.
5. *Гнеденко Б.В., Королюк В.С., Ющенко Е.Л.* Элементы программирования. – М.: Физматгиз, 1961. – 348 с.