S©MR

СИБИРСКИЕ ЭЛЕКТРОННЫЕ
МАТЕМАТИЧЕСКИЕ ИЗВЕСТИЯ
Siberian Electronic Mathematical Reports
http://semr.math.nsc.ru
ISSN 1813-3304

Том 21, № 2, стр. 1370–1384 (2024)
https://doi.org/10.33048/semi.2024.21.086

УДК 510.64, 510,65, 510.6
MSC 03B45, 03Gxx, 03Hxx

МУЛЬТИ-АГЕНТНЫЕ ЛОГИКИ С ВЗАИМОДЕЙСТВИЕМ, УНИФИЦИРУЕМОСТЬ И ПРОЕКТИВНОСТЬ.

В.В. Рыбаков

Представлено С.В.Судоплатовым

Abstract: This paper works with multi-agent none-classical modal logics generated by relational Kripke-like models describing transfer information and its' reliability. We suggest Kripke-like models essentially extending usual multi-modal Kripke semantics. We primarily study algorithmic problems connected with such logics. We find a proof that such logics are decidable, finding algorithms verifying satisfiability formulas, we also solve the problem of admissibility inference rules via technique of projective formulas and unification, we prove that this problem is decidable in such logics.

Keywords: modal logics, multi-agent logics, information, knowledge, problems of unification and admissibility, solving algorithms.

1 Введение

Многоагентные и модальные логики активно использую́сь в проблематике искусственного интеллекта и информатике начиная с 1980гг. Суммирование полученных к начальному времени результатов исследований можно найти в книге [1].

Rybakov V.V., Multi-agent logics with interaction, unifiability and projectivity.

Работа поддержана РНС (23-21-00213).

Поступила 22 сентября 2024 г., опубликована 23 декабря 2024 г.

Мы ранее, анализируя этот подход, предложили его обобщение, используя вместо формулы – знает $K_i A$ – общее определение для информируемости агентов. Эта техника предложена и исследована в Rybakov [2]. В этой работе найдены теоремы о разрешимости таких логик и их приложения. С тех пор было уделено много внимания приложениям много-агентных модальных логик к проблемам построения формальный моделей для концепции знания (см. [2], [3, 4, 5, 6], [6,7, 8–16).

В этой нашей работе мы исследуем много-агентные неклассические модальные логики, порождаемые реляционными Крипке–подобными моделями, описывающими доступ агентов к информации и ее передачу. Эти модели существенно расширяют стандартную семантику Крипке, в них моделируются различные методы доступа агентов к информации. Основная цель – исследование алгоритмических проблем. Мы находим решение проблемы разрешимости для таких логик, находим алгоритмы, проверяющие выполнимость формул. На основе техники проективности формул и унификации решается проблема допустимости правил вывода в таких логиках.

2 Мульти-Агентные Логики с Взаимодействием

Начнем с модификации, предполагающей введение изолированных групп экспертов с редактируемыми правилами доступа к информации. Итак, основным стартовым объектом является линейный временной фрейм

$$F := \langle \bigcup_{i \in N} C(i), \leq \rangle$$

где $\bigcup_{i \in N} C(i)$ – это множество состояний (миров, временных состояний, $i \in N$ – это индексы временный состояний, обозначаемые натуральными числами, $C(i)$ – некоторые множества состояний (элементов, миров), доступных в текущий момент времени i. Для $a \in C(i_1), b \in C(i_2)$, $a \leq b$ если и только если $i_1 \leq i_2$. То есть $\leq$ – это линейное (но не частично упорядоченное) отношение временной достижимости.

Фиксируется некоторое конечное множество индексов J для агентов (можем считать, что $J \subset N$, хотя можем выбрать для обозначения любые символы), и для каждого агента $j \in J$ и каждого $C(i)$ фиксируется некоторое подмножество - группа состояний (миров, элементов, которые агент может запрашивать об истинности утверждений). Основа предлагаемой нами модификации состоит в разделении временных состояний, достигнутых для анализа информации, на более структурированные модели. Рассматриваются собственно сами временные дискретные состояния (такие как сегодня, завтра, послезавтра) и в каждом отдельным временном состоянии рассматриваются формальные модели (модели Крипке), описывающие обмен информацией (моделирующие различные базы данных, такие как веб сайты и пр.) Этот подход изначально может быть проинтерпретирован следующим образом.

Определение 1. Временной многоагентный фрейм – это

$$F := \langle \bigcup_{i \in N} C(i), \leq, R_j, j \in J \rangle,$$

где $i \in N$ и любое $C(i)$ является просто некоторым множеством (называемым интерпретируемым как все информационные базы данных (веб узлы), доступные во время i). При разных i_1 и i_2 множества $C(i_1)$ и $C(i_2)$ предполагаются непересекающимися.

Фиксируется некоторое конечное множество агентов J; любое J разбито на непересекающиеся множества агентов–экспертов, $J = J_1 \cup J_1 \cup \ldots \cup J_k$. Каждому J_m в любом $C(i)$ соответствует некоторое подмножество $Q(J_{i,m})$ множества $C(i)$ и все $Q(J_{i,m})$ также предполагаются непересекающимися. Также предполагаем, что объединение всех $Q(J_{i,m})$ в $C(i)$ дает все множество $C(i)$ (т.е компетенция агентов покрывает все состояния). Бинарные отношения достижимости агентов R_j между состояниями любого множества $C(i)$ для любого $j \in J$ определяются следующим образом

$$\forall a, b \in C(i), aR_jb \iff \exists Q(J_{i,m})[j \in J_m \& (a, b \in Q(J_{i,m}))].$$

То есть, все агенты из J_m могут видеть все состояния из $Q(J_{i,m})$ (внутри любого $C(i)$) и только эти состояния. Это значит, что они обладают абсолютно равными и одинаковыми возможностями для консультации об истинности информации. Это представляется реалистичным и более точным подходом к согласованию истинности информации. А именно, в любой фиксированный момент текущего времени (сегодня, например), для каждого эксперта существует некоторое фиксированное сообщество экспертов с которым эксперт может взаимно обмениваться информацией о том истинно ли утверждение.

Естественно любой фрейм превращается в модель введением означивания логических переменных, а именно

Определение 2. *Модель на фрейме F задается введением некоторого означивания V на F некоторого множества пропозициональных переменных p, где $V(p) \subseteq \bigcup_{i \in N} C(i)$. Все состояния из $V(p)$ трактуются как те, на которых переменные p истинны относительно V.*

Логический язык стандартно включает все булевские логические операции, модальные операции $\Box$ и $\Diamond$ и унарные логические операции K_i для всех $i \in J$, $K_i\varphi$ – означает – агент i знает φ, или агент i может узнать φ (мы не фиксируем жестко трактовку). Отношения, определяющие истинность логических операций перечислены ниже. Пусть задан фрейм $F := \langle \bigcup_{i \in N} C(i), \leq \rangle$ и модель $M := \langle F, V \rangle$ на нем.

Определение 3. $\forall a \in \bigcup_{i\in N} C(i)$: $(F,a) \Vdash_V p \;\Leftrightarrow\; p \in V(p)$;

для всех булевых операций определения вычисления истинности обычные и

$$\forall a \in C(i) \;\; [(F,a) \Vdash_V \Box\varphi \;\Leftrightarrow\; \forall b \in C(j)(i \le j \Rightarrow (F,b) \Vdash_V \varphi)))];$$

$$\forall a \in C(i) \;\; [(F,a) \Vdash_V \Diamond\varphi \;\Leftrightarrow \exists b \in C(j)(i \le j) \& (F,b) \Vdash_V \varphi)).]$$

Истинность логических операций знания K_j определяются так:

$$\forall a \in C(i)\; [(F,a) \Vdash_V K_j\varphi \;\Leftrightarrow \exists b[(aR_jb) \& (F,b) \Vdash_V \varphi)] \vee [(F,a) \Vdash_V \varphi])].$$

При такой интерпретации $K_i\varphi$ означает, что агент может узнать, хотя бы раз, что φ бывает истинно. Далее (уже стандартным образом) по любому фиксированному классу моделей определяются логики, а именно

Определение 4. *Для класса всех фреймов K, определенных выше (для каждого зафиксированного конечного множества агентов J), многоагентная логика $LMA(K)$, порожденная K, – это множество всех формул, истинных на всех состояниях фрейма при всех означиваниях*

Теорема 5. *Логика $LMA(K)$ разрешима, существует алгоритм проверяющий выполнимость формул.*

Доказательство. Итак, дана формула φ и мы должны проверить входит ли эта формула в $LMA(K)$. Заметим, что прямой метод вариаций фильтраций здесь неприменим. Может случится, что в некотором состоянии некоторого $C(i)$ все истинностные значения подформул проверяемой формулы совпадают с истинностными значениями подформул на некоторым состоянии фрейма, стоящим вне $C(i)$ в некотором $C(j)$. Тогда мы будем вынуждены сливать это состояние с некоторым состоянием из $C(j)$ и следовательно пересекать разные компоненты фрейма, что противоречит выбранному семантическому заданию логики. Поэтому необходима предварительная несложная техники отличная от фильтрации (и, далее, использование техники просеивания моделей). Итак, допустим, что формула φ выполняется в некоторой модели

$$(F,V) := \langle \bigcup_{i\in N} C(i), \le, R_j, j \in J, V\rangle$$

на некотором состоянии a, $(F,a) \Vdash_V \varphi$; мы тогда естественно можем предположить, что имеет место $a \in C(0)$.

Первый шаг доказательства состоит в том, что мы изолируем каждое подмножество $Q(J_{i,m})$ в любом временном сгустке $C(i)$ и модифицируем отдельно каждое $Q(J_{i,m})$ (напомним, что они не пересекаются). Вводим отношение $\equiv$ на элементах множества $Q(J_{i,m})$:

$$\forall x, y \in Q(J_{i,m})[x \equiv y \Leftrightarrow [\forall p \in Var(\varphi)[x \Vdash_V p \Leftrightarrow y \Vdash_V p]].$$

Понятно что $\equiv$ является отношением эквивалентности, и мы замещаем любой $x \in Q(J_{i,m})$ классом эквивалентности $[x]_{\equiv}$. Мы оставляем между $[x]_{\equiv}$ все те же отношения достижимости R_j что и были ранее. То есть

$$\forall a, b \in Q(J_{i,m}), [a]_{\equiv} R_j [b]_{\equiv} \iff j \in J_m.$$

Эту процедуру мы выполняем с каждым $Q(J_{i,m})$ в любом сгустке $C(i)$ отдельно (напомним $Q(J_{i,m})$ не пересекаются). Эта процедура не является фильтрацией, но дает нам необходимое свойство: легко проверить индукцией по длине формул, что указанная трансформация сохраняет истинность формул

$$\forall x \in F, \forall \psi \in Sub(\varphi)[(F, x) \Vdash_V \psi \ \Leftrightarrow \ [x]_{\equiv} \Vdash_V \psi].$$

В итоге каждый сгусток $C(i)$ в модели, полученной описанным преобразованием (слиянием элементов в сгустках $C(i)$ исходной модели), становится конечным с числом состояний ограниченным числом вычислимым по размерности множества $Sub(\varphi)$. Далее работаем с только такими моделями и выполняем далее технику просеивания.

А именно, теперь, рассматриваем все из получившихся в результате этой конструкции сгустки $C(i)$ внутри всего фрейма как идентичные если они изоморфны. Пусть

$$\forall n \in N, \ M(n) := \{C(i) \mid i \geq n\}.$$

Ввиду того, что осталось только конечное число таких моделей C(i) и их число ограничено числом вычислимым от числа подформул множества $Sub(\varphi)$, получаем, что существует число $st \in N$, такое что

$$M(st) = M(k), \forall k \geq st.$$

Заметим, что среди временных сгустков $C(i)$, расположенных раньше st, мы можем оставить только вычислимое конечное число сгустков, выбирая максимальные по возможной структуре и расположенные по времени прежде st.

Пусть

$$Repr(st) := \{C(i_1), \ldots, C(i_t)\}$$

– это множество всех возможный различных сгустков из $M(st)$ при $i_j > st$. Порядок и последовательность выбора представителей неважны, но индексы i_j взяты в возрастающем порядке. Далее выберем множество

$$Repr(st2) := \{C(k_1), \ldots, C(k_t)\}$$

которое тоже является множеством всех возможных различных моделей $C(i)$ но уже из множества $M(i_t)$ следующих по времени строго после $C(i_t)$ (снова порядок и последовательность выбора представителей неважны, но тем не менее индексы k_j взяты в возрастающем порядке). Теперь преобразуем получившуюся модель, считая временной индексом следующим за k_t число st и удаляя полностью все сгустки выше времени k_t. Обозначим получившуюся модель через M_1.

Построение модели M_1 закончено и ее структура полностью описана выше (здесь мы опять не используем фильтрацию).

Лемма 6. Для любой формулы ψ из $Sub(\varphi)$ и для любого $a \in M_1$

$$(M_1, a) \Vdash_V \psi \iff (M, a) \Vdash_V \psi.$$

Доказательство следует несложной прямой индукцией по длине формул ψ учитывая выбор чисел st и k_t выше и выбор всех оставшихся в модели сгустков $C(i)$. $\square$

Если же выполняется $(M_1, a) \Vdash_V \varphi$ для некоторой такой конечной модели M_1, то разворачивая цикл сгустков расположенных между $C(st)$ и $C(k_t)$ в возрастающую бесконечную цепь, получаем что $(M, a) \Vdash_V \varphi$ и для некоторой стандартной бесконечной модели M.

Таким образом, благодаря этой лемме, мы сводим вопрос о выполнимости в логике $LMA(K)$, порожденной бесконечными моделями с произвольной структурой моделей, к выполнимости формул на конечных моделях вида M_1 размера не выше вычислимого по размеру формулы. Теорема доказана. $\square$.

Рассмотрим возможный вариант определения локального знания, локальной информации агентов, через предложенные семантические модели. Один из возможных путей выглядит так (итак пусть логическая связка LCK интерпретируется как обозначение для локального знания).

$$\forall a \in C(i) \quad [(F, a) \Vdash_V LCK\varphi \Leftrightarrow [(F, a) \Vdash_V \varphi] \vee$$
$$[\exists j \in J, \exists b \in C(i)(aR_j b \ \& \ (F, b) \Vdash_V \varphi)]].$$

Ясно, что эта связка очевидно выразима в предложенном логическом языке, а именно

$$LCK\varphi \ \equiv \ \varphi \ \vee \bigvee_{j \in J} K_j \varphi.$$

То есть это означает, что либо формула φ истинна уже в текущем состоянии (на a) и существует агентные отношение достижимости из a к состоянию - базе данных – которая знает φ. Локальность здесь означает, что это возможно только внутри текущего кластера времени. $C(i)$. Думается, что такой подход разумен и хорошо совпадает часто с реальной регламентацией работы агентов для получения информации.

Сейчас мы введем еще один возможный способ формализовать локальное общее знание. Идея состоит в том, чтобы не назначать никакой регулярной системы изолированных баз знаний для только экспертов, а регламентировать доступ к информации предписанием агентам администратором. В принципе принятый подход состоит в том, что любому агенту жестко назначено множество состояний и агентов, с которыми он может консультироваться и получать от них информацию. Формально это выглядит следующим образом.

Определение 7. Фрейм внешне выглядит как и в предыдущем определении

$$F := \langle \bigcup_{i \in N} C(i), \leq, R_j, j \in J \rangle.$$

Но в данном случае мы определяем агентные отношения R_j нестандартным образом, чтобы смоделировать идею администрирования. Задаются они следующим образом. Выбрано и зафиксировано натуральное число k – число агентов. Выбирается некоторое зафиксированное конечное множество состояний $a = \{a_1, \ldots, a_k\} \in C(i)$ в моделях (возможны повторения); и для каждого из них зафиксировано некоторое множество состояний $In_a \subseteq C(I)$, с которыми агент a (как состояние) может консультироваться, находясь в состоянии a:

$$R_a := \bigcup \{ < a, b > \mid b \in In_a \} \cup \{ < a, a > \}.$$

То есть теперь поведение R_a не регулярно и полностью определено администратором (хотя думается, это хорошо соответствует решениям, принятым реальным администраторам). Помимо этого, мы предполагаем, что различные множества R_a полностью не пресекаются – не имеют общих состояний вообще, ни одно состояние b модели не может входить как компонента бинарных отношений $< x, y >$ в разных R_a.

Для моделей на таких фреймах мы таким же как и ранее образом определяем вычисление истинности логических операций, в частности K_a. Как и ранее логика вводится обычным образом.

Определение 8. *Для класса K указанных фреймов, многоагентная логика LRI порожденная K – это множество всех формул, истинных на всех состояниях таких фреймов при всех возможных означиваниях.*

Теорема 9. *Логика LRI разрешима, существует алгоритм, проверяющий выполнимость формул.*

Доказательство. Мы вначале изолируем каждое множество

$$R_a := \bigcup \{ < a, b > \mid b \in In_a \} \cup \{ < a, a > \}$$

и выполняем на нем следующие преобразования. Пусть

$$\forall x, y \in In_a [xEy \ \Leftrightarrow \ [\forall p \in Var(\varphi)[x \Vdash_V p \Leftrightarrow y \Vdash_V p]].$$

Очевидно, что это отношение E является отношением эквивалентности на In_a, и мы берем фактор множество на In_a по E и заменяем все $b \in In_a$ на содержащие их фактор множества $[b]_E$ и полагаем $[a]_E := \{a\}$, далее считаем $[a]_E R_a [b]_E$. Выполняем такое преобразование для каждого $C(i)$

и для каждого R_a при a из $C(i)$. Получаем модель M_1. В M_1 очевидно выполняется

$$\forall x \in F, \forall \psi \in Sub(\varphi)[(F,x) \Vdash_V \psi \;\; \Leftrightarrow \;\; (M_1, [x]_E) \Vdash_V \psi],$$

где $[x]_E$ – это состояние, соответствующее x после сделанного преобразования.

После этого в любом сгустке $C(i)$ получившейся модели из множества всех получившихся R_a, изоморфных между собой как модели с означиванием, удалим все копии, оставив только одну такую уникальную модель R_a. В полученной таким образом модели M_2 очевидно тоже выполняется

$$\forall x \in M_1, \forall \psi \in Sub(\varphi)[(M_1,x) \Vdash_V \psi \;\; \Leftrightarrow \;\; (M_2, x_s) \Vdash_V \psi],$$

где x_s – это состояние, соответствующее x после сделанного преобразования.

После этого получаем, что любое $C(i)$ конечно и имеет число элементов не выше числа эффективно вычислимого по числу подформул формулы φ.

Теперь, рассматриваем все из получившихся в результате этой конструкции модели $C(i)$ внутри всего фрейма как идентичные, если они изоморфны. Пусть $\forall n \in N,\; M(n) := \{C(i) \mid i \geq m\}$. После этого осталось только конечное число таких моделей C(i).

Ясно, что их число ограничено числом вычислимым из числа подформул нашей формулы. Поэтому получаем, что существует число $st \in N$, такое что $M(st) = M(k), \forall k \geq st$. Как мы уже использовали ранее среди временных сгустков $C(i)$, расположенных раньше st, мы можем оставить только вычислимое конечное число сгустков, выбирая максимальные по возможной структуре и расположенные по времени прежде st. Пусть как и ранее $Repr(st) := \{C(i_1, \ldots, C(i_t)\}$ – это множество всех возможный различных моделей такого сорта из $M(st)$ при $i_j > st$. Порядок и последовательность выбора представителей неважны, но индексы i_j взяты в возрастающем порядке. Далее мы снова выберем множество $Repr(st_F) := \{C(k_1, \ldots, C(k_t)\}$, которое тоже является множеством всех возможный различных моделей $C(i)$, но уже из $M(i_t)$ при $k_j > i_t$. (снова порядок и последовательность выбора представителей неважны, но индексы k_j взяты в возрастающем порядке).

Теперь преобразуем получившуюся модель, считая временным индексом следующим за k_t число st и удаляя полностью все сгустки выше времени k_t. Обозначим получившуюся модель через M_3.

Лемма 10. Для любой формулы ψ из $Sub(\varphi)$ и для любого $a \in M_2$

$$(M_2, a) \Vdash_V \psi \iff (M_3, a) \Vdash_V \psi.$$

Доказательство следует несложной прямой индукцией по длине формул ψ учитывая выбор чисел st и k_t выше и выбор всех оставшихся в модели сгустков $C(i)$. $\Box$

В случае, когда выполняется $(M_3, a) \Vdash_V \varphi$ для некоторой такой конечной модели (M_3), то разворачивая цикл сгустков, расположенных между (C_{st}) и (C_{k_t}) в возрастающую бесконечную цепь, получаем, что $(M, a) \Vdash_V \varphi$ и для некоторой стандартной бесконечной модели M.

Таким образом, благодаря этой лемме, мы сводим вопрос о выполнимости в логике LRI, порожденной бесконечными моделями с произвольной структурой моделей, к выполнимости формул на конечных моделях вида M_3 размера не выше вычислимого по размеру формулы. Теорема доказана. $\Box$

3 Вводная информация по унифицируемости

Кратко наполним,что подстановка для множества пропозициональных переменных (или букв) P – это некоторое отображение ε множества P во множество всех формул For. Любая такая подстановка может быть расширена на множество всех формул с посредством $\varepsilon(\varphi(x_1, \ldots, x_n)) := \varphi(\varepsilon(x_1), \ldots, \varepsilon(x_n))$.

Определение 11. Формула φ называется унифицируемой в логике L, если существует подстановка ε (называемая унификатором формулы φ) такая, что $\varepsilon(\varphi) \in L$.

Определение 12. *Унификатор ε для формулы φ в логике L является более общим чем унификатор ε_1 если существует подстановка δ такая, что для любой буквы x, $[\varepsilon_1(x) \equiv \delta(\varepsilon(x))] \in L$.*

Если логика разрешима, то обычно проверить унифицируемость формул довольно просто (теоретически, но сложность и длина вычислений может быть недопустимо высокой) : достаточно использовать только подстановки переменных в – истина-ложна –$\{\bot, \top\}$. Но проблема нахождения всех унификаторов гораздо более сложна.

Определение 13. Множество унификаторов CU для данной формулы φ в логике L является полным множеством унификаторов, если выполняется следующее. Для любого унификатора σ формулы φ в L, существует унификатор σ_1 из CU, такой что σ_1 более общий чем σ.

Для модальных логик, расширяющих модальную систему $S4$ определение проективности формул может быть дано следующим образом.

Определение 14. Формула φ называется проективной в логике L, если выполняется следующее. Существует подстановка σ (называемая проективной подстановкой для φ) такая что σ является унификатором для φ в L и $\Box\varphi \to [x_i \equiv \sigma(x_i)] \in L$ для любой переменной x_i из φ.

Лемма 15. Если подстановка σ_p проективна для φ в логике L, то $\{\sigma_p\}$ является полным множеством унификаторов для φ (т.е. подстановка σ_p является наиболее общим унификатором).

Доказательство. Действительно, пусть σ является унификатором для φ в L. Так как σ_p проективна для φ в L, мы получаем $\Box\varphi \to [x_i \equiv \sigma_p(x_i)] \in L$ для любой переменной x_i из φ. Применяя σ к формулам выше, мы получаем $\sigma(\Box\varphi) \to [\sigma(x_i) \equiv \sigma(\sigma_p(x_i))] \in L$, то есть $\sigma(x_i) \equiv \sigma(\sigma_p(x_i)) \in L$. Q.E.D.

Конечно не все унифицируемые формулы проективны в неклассических логиках, примеры широко известны.

Легко видеть, что если для разрешимой модальной логики L, расширяющей $S4$ мы можем определить проективна ли формула φ и вычислить ее проективный унификатор, то проблема допустимости правил вывода в L разрешима. Действительно, правило вывода $\varphi_1, \dots \varphi_n/\psi$ не допустимо в L если и только если существует унификатор для $\varphi_1 \wedge \dots \wedge \varphi_n$, который не является унификатором для ψ. Но тогда вычислимый более общий унификатор, получаемый по проективной подстановке для $\varphi_1 \wedge \dots \wedge \varphi_n$ проверяет это. Эта интересная идея была введена в обиход логиков, работающих в неклассической логике, Silvio Ghilardi (см. [18]). Мы применим в этой статье технику проективности для решения проблемы допустимости в изучаемых в этой статье логиках.

4 Разрешимость Допустимости через Проективность

Применение проективности в данном конкретном случае логик близких к линейным, позволит дать нам очень короткое и компактное доказательство решения проблемы допустимости правил вывода. Такой подход через технику проективности был найден Сильвио Гиларди и впоследствии был применен к линейным модальным логикам (Wojciech Dzik, Piotr Wojtylak) для модальных логик, расширяющих S.4.3 [20], [18,19] и затем был развит для временных логик в V. Rybakov [17]. В этом параграфе нам необходима некая легкая модификация многоагентных фреймов, для того чтобы мы могли использовать технику проективных формул.

Определение 16. Как и ранее, в качестве фрейма мы можем использовать любую версию определенную ранее —

$$\mathbf{F} := \langle \bigcup_{\mathbf{i} \in \mathbf{N}} \mathbf{C(i)}, \leq, \mathbf{R_j}, \mathbf{j} \in \mathbf{J} \rangle.$$

Но для унификации через проективность нам необходимо присутствие некоторого состояния в каждом кластере $C(i)$: мы допускаем, что для любого $C(i)$ существует по крайней мере единственное состояние $Dis \in C(i)$ такое что,

$\forall i \in N, \forall j \in J, \forall b \in C(i), [DisR_jb \iff b = Dis]\&[bR_jDis \iff b = Dis]$.

Этот элемент Dis называем от-соединенным или выключенным (disjoint), так как любой агент не может его достичь по R_j и снять с него информацию, и также будучи в любом другом состоянии не видит по R_j этот элемент (он как бы невидим агентам). В частности, мы полагаем, что в любой модели при любом выбранном означивании на таком фрейме все переменные на элементах типа Dis ложны ($(M, Dis) \Vdash_V \neg p_j$). Эти ограничения, тем самым, никак не ограничивают выразимость информации, но технически полезны для применения проективности. Как и ранее логика L – это множество всех формул истинных на всех таких возможных моделях.

Теорема 17. *В логике L любая унифицируемая формула является проективной. Унифицируемость формул в этой логике алгоритмически распознаваема. Существует алгоритм, строящий проективный унификатор для любой унифицируемой формулы. Поэтому проблема допустимости правил вывода разрешима для L.*

Доказательство. Вначале нужно доказать алгоритмическую распознаваемость унифицирумых формул. Эта задача не так тривиальна, как для просто булевой пропозициональной логики, но сравнительно достаточно несложна. Действительно, пусть дана произвольная формула $\varphi(x_1, ..., x_n)$ и она унифицируема. Наша логика обладает в частности моделями M, где каждый кластер $C(i)$ состоит только из одного состояния Dis.

Допустим, что формула $\varphi(x_1, ..., x_n)$ унифицируема в логике L и σ_1 ее унификатор. Пусть подстановка σ_1 действует следующим образом,

$$\forall x_i, \sigma_1(x_i) := \alpha_i, \text{ где } \alpha_i \in For$$

При этой подстановке формула $\varphi(x_1, ..., x_n)$ истинна на M, так как σ_1 ее унификатор. Тогда истинность любых переменных y_j из формул в указанных формулах α_i, участвующих в подстановке σ_1, принимают истинностное значение ложь на любом состоянии Dis из этой модели M, – поэтому мы можем считать, что

$$\sigma_1(y_j) = \perp$$

и при этой подстановке σ_1 на этой модели истинна наша формула. При этом мы не утверждаем, что модифицированная подстановка – унификатор в логике, но тем не менее очевидно верна

Лемма 18. Для данной модифицированной подстановки σ_1

$$\forall a \in M, (M, a) \Vdash_V \neg\sigma_1(y_j).$$

В этом случае любая из формул α_i при этой подстановке σ_1 принимает одно и то же истинностное значение истина или ложна одновременно на всех состояниях из этой модели (а она состоит только из таких состояний Dis). Поэтому (легкая индукция по длине формул α_i) верна

Лемма 19. В модели M,

$$\forall a, b \in M,\ \forall \alpha_i (M, a) \Vdash_V \sigma_1(\alpha_i) \Leftrightarrow (M, b) \Vdash_V \sigma_1(\alpha_i).$$

Потому подстановка σ_1 вместо переменных формулы $\varphi(x_1, ..., x_n)$ просто истинностных значений формул α_i в указанной модели M при том заданном означивании σ_1, будет унификатором формулы $\varphi(x_1, ..., x_n)$ в нашей логике (так как $\varphi(x_1, ..., x_n)$ при σ_1 истинна на исходной модели M). Поэтому и здесь для проверки унифицируемости достаточно рассмотреть просто одноэлементную модель и истинность на ней.

Конечно мы не знаем точного значения действия σ_1 на одноэлементной модели (как и в булевой логике), но его можно уточнить простым перебором пользуясь разрешимостью логики (она доказывается как и ранее в этой статье для упомянутых логик, просто в начале доказательства надо слить все состояния типа Dis в любом кластере C(i) в одно состояние). Таким образом проблема распознавания унифицируемости в таких логиках разрешима. Зафиксируем необходимый нам в будущем некоторый вычисленный таким образом унификатор σ_3 для формулы $\varphi(x_1, ..., x_n)$,

$$\sigma_3(x_i) = \delta_i, \text{где } \delta_i \in \{\bot, \top\}.$$

Но вот проблема нахождения наиболее общих унификаторов совсем не так проста.

Приступая к доказательству проективности унифицирумых формул, для любой пропозициональной переменной x_i из любой данной формулы φ мы определим следующую подстановку. Пусть $Var(\Box\varphi)$ является множеством всех пропозициональных переменных из $\Box\varphi$ и $Sub(\Box\varphi)$ является множеством всех подформул формулы $\Box\varphi$. Возьмем любое $X \subseteq Sub(\Box\varphi)$, и далее фиксируем формулы.

Пусть для любой переменной x_i из $Var(\varphi)$,

$$\sigma(x_i) := [(\Box\varphi) \wedge x_i] \vee [\neg\Diamond\Box\varphi \wedge \sigma_3(x_i)] \vee$$

$$[\neg\Box\varphi \wedge \Diamond\Box\varphi \wedge \bot].$$

Докажем, что эта подстановка является унификатором для формулы φ в логике L. Выбираем любую модель нашей логики, т.е. любой фрейм F (с состояниями Dis в любом сгустке) с любым означиванием V для переменных x_i из данной формулы φ. Берем любое состояние $a \in F$.

Рассмотрим случай когда $(F, a) \Vdash_V \neg\Diamond\Box\varphi$. По определению подстановки σ_3, она является унификатором для формулы φ в логике, поэтому

$\forall b \in F,\ (F,b) \Vdash_V \sigma_3(\varphi))$ и соответсвенно $\forall b \in F,\ (F,b) \Vdash_V \sigma(\varphi))$.

Допустим теперь что $(F,a) \Vdash_V \Box\varphi$. Тогда подстановка σ не меняет истинностных значений переменных x_i в сравнении с их истинностными значениями при оригинальном означивании V. Поэтому

$$(F,a) \Vdash_V x_i \Leftrightarrow (F,a) \Vdash_V \sigma(x_i),\ \forall b \geq a\ (F,b) \Vdash_V x_i \Leftrightarrow (F,b) \Vdash_V \sigma(x_i),$$

следовательно

$$(F,a) \Vdash_V \sigma(\Box\varphi).$$

Осталось рассмотреть случай когда

$$(F,a) \Vdash_V \neg\Box\varphi \wedge \Diamond\Box\varphi.$$

Пусть $C(i)$ – это наименьший сгусток, где формула $\Box\varphi$ истинна при оригинальном означивании V, пусть $Dis(i)$ и $Dis(i-1)$ – это два выбранных любых состояния типа Dis из $C(i)$ и $C(i-1)$ соответственно. Тогда для любой $x_i \in Var(\varphi)$ ввиду того, что состояния $Dis(i)$ и $Dis(i-1)$ являются выключенными имеет место

$$(F, Dis(i)) \Vdash_V \neg x_i \&\ (F, Dis(i-1)) \Vdash_V \neg x_i.$$

Используя это и вновь выключенность состояний $Dis(i)$ и $Dis(i-1)$ (ограничения на работу агентных операций R_j) прямой индукцией по длине формулы легко проверить, что для любой булевой формулы α построенной из переменных x_i

Лемма 20.

$$(F, Dis(i)) \Vdash_V \alpha \Leftrightarrow (F, Dis(i-1)) \Vdash_V \alpha.$$

Кроме того верна

Лемма 21. Для любого состояния g типа Dis из любого сгустка $C(j)$ модели и любого означивания V_2 на M, и любой формулы β из $Sub(\Box\varphi)$

$$(F,g) \Vdash_{V_2} \bigwedge_{\beta \in Sub(\Box\varphi)} ([\beta \to \bigwedge_{j \in J} K_j\beta] \wedge [\bigwedge_{j \in J \in Sub(\Box\varphi)} [K_j\beta \to \beta]).$$

Доказательство прямо следует из определения агентных отношений достижимости R_j на состояниях типа Dis.

Из этих двух лемм, прямой индукцией по длине формул α выводится

Лемма 22. Для любой формулы α из $Sub(\Box\varphi)$, построенной из переменных x_i

$$(F, Dis(i)) \Vdash_V \sigma(\alpha) \Leftrightarrow (F, Dis(i-1)) \Vdash_V \sigma(\alpha).$$

Для сгустков $C(j)$ расположенных ниже $C(i-1)$ доказательство предыдущей леммы аналогичное, так как в таких сгустках при подстановке σ переменные имеют то же означивание после применения подстановки σ как и в $C(i-1)$.

Тем самым мы доказали, что подстановка σ, построенная нами и является унификатором для φ. Проективность этой подстановки очевидна.

Теорема доказана

References

[1] R. Fagin, J.Y. Halpern, Y. Moses, M.Y. Vardi, *Reasoning about knowledge*, MIT, Cambridge, 1995. Zbl 0839.68095

[2] V.V. Rybakov, *Refined common knowledge logics or logics of common information*, Arch. Math. Logic, **42**:2 (2003), 179–200. Zbl 1030.03015

[3] S. Artemov, *Evidence-based common knowledge*, Technical Report TR-2004018 CUNY Ph.D. Program in Computer Science, City University of New York (CUNY), 2004.

[4] S. Artemov, *Explicit generic common knowledge*, In: Artemov, S., Nerode, A. (eds), *Logical Foundations of Computer Science. LFCS 2013*, Lecture Notes in Computer Science, **7734**, Springer, Berlin, Heidelberg, 2013.

[5] S. Artemov, *Justification awareness*, J. Log. Comput., **30**:8 (2020), 1431–1446. Zbl 1496.03068

[6] V. Rybakov, *Temporal Multi-agent's logic, knowledge, uncertainty, and plausibility*, In: Jezic, G. (ed.) et al., *Agents and Multi-Agent Systems: Technologies and Applications 2021*, Smart Innovation, Systems and Technologies, **241**, Springer, Singapore, 2021, 205–214.

[7] F. Baader, R. Kusters, *Unification in a description logic with transitive closure of roles*, In Nieuwenhuis, Robert (ed.) et al., *Logic for programming, artificial intelligence, and reasoning*, LNCS, **2250**, Springer, Berlin, 2001, 217–232. Zbl 1275.68134

[8] V.V. Rybakov, *Problems of substitution and admissibility in the modal system Grz and in intuitionistic propositional calculus*, Ann. Pure Appl. Logic, **50**:1 (1990), 71–106. ZBL 0709.03009

[9] V.V. Rybakov, *Rules of inference with parameters for intuitionistic logic*, J. Symb. Log., **57**:3 (1992), 912–923. ZBL 0788.03007

[10] V.V. Rybakov, *Logical consecutions in discrete linear temporal logic*, J. Symb. Log., **70**:4 (2005), 1137–1149. ZBL 1110.03010

[11] V.V. Rybakov, *Linear temporal logic with until and next, logical consecutions.* Annals of Pure and Applied Logic, 155 (2008), 32 – 45.

[12] V.V. Rybakov, *Projective formulas and unification in linear temporal logic* LTL_U, Log. J. IGPL, **22**:4 (2014), 665–672. ZBL 1342.03017

[13] V.V. Rybakov, *Non-transitive linear temporal logic and logical knowledge operations*, J. Log. Comput., **26**:3 (2016), 945–958. ZBL 1403.03028

[14] V.V. Rybakov, *Nontransitive temporal multiagent logic, information and knowledge, deciding algorithms*, Sib. Math. J., **58**:5 (2017), 875–886. ZBL 1420.03060

[15] V.V. Rybakov, *Multiagent temporal logics with multivaluations*, Sib. Math. J., **59**:4 (2018), 710–720. ZBL 1469.03047

[16] V.V. Rybakov, *Branching time agents logic, satisfiability problem by rules in reduced form*, Sib. Èlektron. Mat. Izv., **16** (2019), 1158-–1170. ZBL 1436.03126

[17] S. Ghilardi, *Unification through projectivity,* J. Log. Comput., **7**:6 (1997), 733–752. ZBL 0894.08004

[18] S. Ghilardi, *Unification, finite duality and projectivity in varieties of Heyting algebras,* Ann. Pure Appl. Log., **127**:1-3 (2004), 99–115. ZBL 1058.03020

[19] W. Dzik, P. Wojtylak, *Projective unification in modal logic*, Log. J. IGPL, **20**:1 (2012), 121–153. ZBL 1260.03041

Rybakov Vladimir Vladimirovich
Siberian Federal University,
79 Svobodny pr.,
660041, Krasnoyarsk, Russia
Email address: vrybakov@sfu-kras.ru