

УДК 519.767.2
DOI: 10.15827/0236-235X.125.020-025

Дата подачи статьи: 27.12.18
2019. Т. 32. № 1. С. 020–025

Унифицированное представление формул логик LTL и CTL системами рекурсивных уравнений

Ю.П. Кораблин^{1,2}, д.т.н., профессор, y.p.k@mail.ru

А.А. Шипов³, к.т.н., старший инженер-программист, a-j-a-1@yandex.ru

¹ Российский технологический университет (МИРЭА), г. Москва, 119454, Россия

² Национальный исследовательский университет «МЭИ», г. Москва, 111250, Россия

³ РСК Технологии, г. Москва, 121170, Россия

Для решения задачи верификации методом проверки на моделях Model Checking сегодня зачастую используются такие временные логики, как логика линейного времени LTL, логика ветвящегося времени CTL и логика CTL*, объединяющая возможности двух первых логик. Однако каждая из этих логик имеет свои недостатки, ограничения и проблемы выразительности, которые возникают ввиду их синтаксических и семантических особенностей. Именно поэтому на текущий момент не существует единой темпоральной логики.

Авторы данной статьи убеждены, что использование специальных представлений, основанных на системах рекурсивных уравнений в отношении темпоральных логик, способно не только расширить их выразительную мощь, но и унифицировать синтаксические конструкции, позволив тем самым сформулировать некоторую общую и единую для всех логик нотацию.

В статье предложена и рассмотрена специальная RTL-нотация, в основе которой лежат системы рекурсивных уравнений и привычные семантические определения логик LTL и CTL. Задача, которую призвана решить данная нотация, состоит в объединении выразительных возможностей обеих логик, что расширит выразительность каждой из них, а также в унификации их синтаксических конструкций, что даст возможность выработать единообразный подход к решению задачи верификации.

Авторами дано подробное определение RTL-нотации, представлены соответствующие аксиомы и теоремы, приведен ряд примеров и утверждений, наглядно демонстрирующих выразительные возможности RTL.

Целью статьи является демонстрация ключевых особенностей и возможностей RTL-нотации, которые в дальнейших работах авторов лягут в основу решения проблемы верификации моделей систем.

Ключевые слова: верификация, Model Checking, эквивалентная характеристика RLTL, LTL, CTL, формула временной логики.

Верификация сложных программных и технических систем давно стала неотъемлемой частью их жизненного цикла. Существующие инструменты и средства верификации постоянно развиваются и совершенствуются с целью повышения их быстродействия, гибкости и расширения области применения. Так, одним из наиболее актуальных на сегодняшний день является метод формальной верификации на моделях, или Model Checking [1, 2].

Концептуальная особенность данного метода в том, что верификация выполняется путем анализа свойств моделей систем, построенных на базе некоторых формальных конструкций, относительно требований к ним, также заданных формально. Как правило, для задания требований оперируют такими временными логиками, как LTL, CTL, CTL* [3–6]. Однако выразительные способности этих логик не являются безграничными и в некоторых случаях невозможно задать с их помощью необходимые требования.

Авторами в ряде предыдущих работ уже были предприняты некоторые меры, способные значительно расширить выразительную способность логики линейного времени LTL путем использования специальной RLTL-нотации, описывающей формулы LTL в виде систем рекурсивных уравнений [7]. Помимо этого, было показано, что при помощи RLTL можно задавать не только сами требования, но и модели вери-

фицируемых систем, что позволяет упростить процесс верификации и повысить его эффективность [8].

Логика CTL [6] является альтернативным по отношению к LTL способом формулирования временных свойств. Если в логике линейного времени свойства формулируются лишь относительно вычислительного процесса, то в логике ветвящегося времени, помимо порядка, учитываются также сами альтернативы развития вычислительного процесса. В отличие от логики линейного времени, формулы которой являются формулами пути, формулы логики ветвящегося времени являются формулами состояний. То есть любой темпоральный оператор формулы CTL предварен квантором пути, который определяет, на всех (квантор всеобщности A) или только на некоторых (квантор существования E) путях из текущего состояния соответствующая формула является истинной [1].

Верифицируемая система может работать в разных режимах в зависимости от условий или исходных данных, может поддерживать многопоточное/многопроцессное выполнение [9] или результаты работы системы носят вероятностный характер [10]. Во всех этих ситуациях логики линейного времени, скорее всего, будет недостаточно.

В данной статье рассматриваются проблемы выразительности формул логики ветвящегося времени, или

CTL. В статье наглядно продемонстрировано, что добавления нескольких синтаксических конструкций в нотации RLTL достаточно, чтобы выразить формулы логики не только линейного, но и ветвящегося времени.

RTL-нотация

Предложенная авторами в работе [6] RLTL-нотация позволяет представлять операторы LTL и их композиции в виде систем рекурсивных уравнений (см. табл. 1, 2).

Таблица 1

Рекурсивные представления операторов LTL в RLTL

Table 1

Recursive representations of LTL operators in RLTL

LTL	RLTL
$F\phi = \phi \vee XF\phi$	$F = \phi + \Delta \circ F$
$G\phi = \phi \wedge XG\phi$	$G = \phi \circ G$
$\phi_1 U \phi_2 = \phi_2 \vee \phi_1 \wedge X(\phi_1 U \phi_2)$	$U = \phi_2 + \phi_1 \circ U$
$\phi_1 \wedge X\phi_2$	$\phi_1 \circ \phi_2$
$X\phi$	$\Delta \circ \phi$

Таблица 2

Примеры представления композиций операторов LTL в RLTL

Table 2

The examples of composition representations of LTL operators in RLTL

LTL	RLTL
$FG\phi$	$F = \phi \circ G + \Delta \circ F$ $G = \phi \circ G$
$GF\phi$	$G = \phi \circ G + \Delta \circ G$

Формулам LTL в RLTL-нотации соответствуют системы рекурсивных уравнений. В левой части каждого уравнения записывается метaperменная, уникально идентифицирующая данное уравнение, в правой – темпоральное выражение, содержащее как метaperменные, так и терминальные символы (предикаты).

Замечание. Не следует путать использование метaperменных F, G и U в рекурсивных уравнениях с одноименными темпоральными операторами.

Символ \circ обозначает оператор продолжения (конкатенация выражений), что позволяет использовать оператор X в неявном виде, а также упростить его восприятие, в частности, запись $\phi_1 \circ \phi_2$ будем понимать как ϕ_2 следует за ϕ_1 . Символом + соединяются альтернативные ветви выражений. Под символом Δ , именуемым предикатом неопределенности, будем понимать некоторое неопределенное подмножество символов входного алфавита, под отрицанием данного предиката – некоторое другое неопределенное подмножество символов алфавита, а под отрицанием отрицания – некоторое третье неопределенное подмножество. Таким образом, отрицание предиката неопределенности является неполным, а каждое последующее отрицание дает неопределенное подмножество символов. Каждое из этих подмножеств ввиду

своей неопределенности также может быть обозначено через Δ . Таким образом, мы имеем следующее свойство: $\neg\Delta = \Delta$.

Запись вида Δ^ω является расширением предиката неопределенности и представляет собой сокращенную запись для рекурсивного уравнения $F = \Delta \circ F$, означающего наступление бесконечной последовательности предикатов неопределенности на всех путях вычислительного процесса, исходящих из данного состояния. При этом следует отметить, что наступление предиката неопределенности в некотором состоянии вычислительного процесса не отрицает наступление в этом же состоянии любого из символов входного алфавита.

Унифицированная нотация логик LTL и CTL, далее именуемая RTL-нотацией, представляет собой RLTL-нотацию, расширенную и дополненную рядом аксиом и условных обозначений формул логики ветвящегося времени. Для расширения RLTL-нотации в RTL следует ввести следующие допущения:

- формулы, заданные с помощью системы рекурсивных уравнений RLTL, по умолчанию представляют собой формулы логики ветвящегося времени относительно всех ветвей вычислительного процесса, исходящих из текущего состояния;

- для задания формулы ветвящегося времени относительно лишь некоторых ветвей вычислительного процесса необходимо в самой формуле явным образом описать все альтернативы вычислительного процесса.

В таблице 3 представлены все основные операторы логики CTL и соответствующие им синтаксические конструкции RTL.

Таблица 3

Рекурсивные представления операторов CTL в RTL-нотации

Table 3

Recursive representations of CTL operators in RTL notation

№	CTL	RTL
1	$AX\phi$	$\Delta \circ \phi$
2	$EX\phi$	$\Delta \circ (\phi + \Delta)$
3	$AF\phi = \phi \vee AX AF\phi$	$F = \phi + \Delta \circ F$
4	$EF\phi = \phi \vee EX EF\phi$	$F = \phi + \Delta \circ (F + \Delta^\omega)$
5	$AG\phi = \phi \wedge AX AG\phi$	$F = \phi \circ F$
6	$EG\phi = \phi \wedge EX EG\phi$	$F = \phi \circ (F + \Delta^\omega)$
7	$A(\phi_1 U \phi_2) = \phi_2 \vee \phi_1 \wedge AX A(\phi_1 U \phi_2)$	$F = \phi_2 + \phi_1 \circ F$
8	$E(\phi_1 U \phi_2) = \phi_2 \vee \phi_1 \wedge EX E(\phi_1 U \phi_2)$	$F = \phi_2 + \phi_1 \circ (F + \Delta^\omega)$

Обозначим через $\mathcal{P}(C)$ множество всех последовательностей предикатов, соответствующих формуле C логики ветвящегося времени, а через $\mathcal{P}(R(C))$ – множество всех последовательностей предикатов, соответствующих представлению формулы C в RTL.

Теорема 1. Для всех формул C логики ветвящегося времени $\mathcal{P}(C) = \mathcal{P}(R(C))$.

Доказательство. Формула $AX\phi$ означает, что, каким бы ни было значение предиката на данном шаге, на следующем шаге будет истинным значение предиката ϕ . Запись $\Delta \circ \phi$ означает, что на данном шаге выполняется некоторое неопределенное множество пре-

дикатов (возможно, пустое), а на следующем шаге будет истинным значение предиката φ , что в точности совпадает со смыслом формулы $AX\varphi$.

Рассмотрим также доказательство теоремы для формулы $EG\varphi$. В соответствии с семантикой этой формулы в CTL множество последовательностей включает в себя, по крайней мере, одну ветвь, для которой на данном шаге справедливо утверждение (предикат) φ , а на каждом следующем шаге, опять же хотя бы в одной ветви, являющейся продолжением данной ветви, справедливо утверждение φ . Из представления данной формулы в RTL нетрудно понять, что на данном шаге хотя бы в одной ветви должно быть справедливым утверждение φ , а на каждом следующем шаге, опять же хотя бы в одной ветви, являющейся продолжением данной ветви, справедливо утверждение φ , что в точности совпадает со смыслом формулы $EG\varphi$.

Аналогичным образом можно показать и справедливость всех приведенных в таблице 3 представлений формул CTL.

Если сделать предположение о том, что бесконечно повторяющаяся альтернатива должна быть когда-нибудь выбрана, то нетрудно осуществить и обратный переход от представлений формул в RTL к исходным формулам CTL.

Отметим, что RTL представляет собой именно нотацию для представления формул логик LTL и CTL, поскольку базируется на их семантических определениях, но никак не отдельную логику.

Сформулируем аксиомы и правила вывода нотации RTL, задающие систему эквивалентных преобразований формул CTL. Символом γ (с индексами и без) будем обозначать темпоральные выражения.

Аксиомы RTL:

- A1. $\neg F\gamma = G\neg\gamma$
- A2. $\neg G\gamma = F\neg\gamma$
- A3. $\neg A\gamma = E\neg\gamma$
- A4. $\neg E\gamma = A\neg\gamma$
- A5. $\neg\Delta = \Delta$
- A6. $\Delta^\omega \circ \gamma = \Delta^\omega$
- A7. $\Delta^\omega = \Delta \circ \Delta^\omega$
- A8. $\gamma_1 + \gamma_2 = \gamma_2 + \gamma_1$
- A9. $\gamma + \gamma = \gamma$
- A10. $(\gamma_1 + \gamma_2) + \gamma_3 = \gamma_1 + (\gamma_2 + \gamma_3)$
- A11. $\gamma_1 \circ (\gamma_2 \circ \gamma_3) = (\gamma_1 \circ \gamma_2) \circ \gamma_3$
- A12. $(\gamma_1 + \gamma_2) \circ \gamma_3 = \gamma_1 \circ \gamma_3 + \gamma_2 \circ \gamma_3$
- A13. $\gamma_1 \circ (\gamma_2 + \gamma_3) = \gamma_1 \circ \gamma_2 + \gamma_1 \circ \gamma_3$

Правила вывода RTL:

- R1. $F = \gamma_1 \circ F \circ \gamma_2 \rightarrow F = \gamma_1 \circ F$
- R2. $F = \gamma_1 \circ F \rightarrow F = \gamma_1 \circ F \circ \gamma_2$ (и в обратную сторону правило справедливо)

Сочетания операторов CTL

Сочетания операторов логики ветвящегося времени, заданных в своем рекурсивном представлении, также могут быть легко получены на основе обозначенных в предыдущем разделе аксиом и правил вы-

вода. Рассмотрим некоторые из них и опишем процесс их вывода.

Утверждение 1. Выражение вида $AGAF\varphi$ логики CTL может быть представлено в RTL-нотации в виде $F = \varphi \circ F + \Delta \circ F$.

Доказательство. Обозначим через F выражение $AGAF\varphi = AG(AF\varphi)$, а через $F1$ выражение $AF\varphi$.

$$F = AGAF\varphi = AG(AF\varphi) = (\text{п. 5 табл. 3})$$

$$F1 \circ F = (\text{п. 3 табл. 3})$$

$$(\varphi + \Delta \circ F1) \circ F = (A12)$$

$$\varphi \circ F + \Delta \circ F1 \circ F = (A11)$$

$$\varphi \circ F + \Delta \circ (F1 \circ F) =$$

$$\varphi \circ F + \Delta \circ F, \text{ ч.т.д.}$$

Утверждение 2. Выражение вида $AGEF\varphi$ логики CTL может быть представлено в RTL-нотации в виде $F = \varphi \circ F + \Delta \circ (F + \Delta^\omega)$.

Доказательство. Обозначим через F выражение $AGEF\varphi = AG(EF\varphi)$, а через $F1$ выражение $EF\varphi$.

$$F = AGEF\varphi = AG(EF\varphi) = (\text{п. 5 табл. 3})$$

$$F1 \circ F = (\text{п. 4 табл. 3})$$

$$(\varphi + \Delta \circ (F1 + \Delta^\omega)) \circ F = (A12)$$

$$\varphi \circ F + \Delta \circ (F1 \circ F + \Delta^\omega \circ F) = (A6)$$

$$\varphi \circ F + \Delta \circ (F + \Delta^\omega), \text{ ч.т.д.}$$

Утверждение 3. Выражение вида $AFAG\varphi$ логики CTL может быть представлено в RTL-нотации в виде системы рекурсивных уравнений:

$$F = \varphi \circ F1 + \Delta \circ F$$

$$F1 = \varphi \circ F1$$

Доказательство. Обозначим через F выражение $AFAG\varphi = AF(AG\varphi)$, а через $F1$ выражение $AG\varphi$.

$$F = AFAG\varphi = AF(AG\varphi) = (\text{п. 3 табл. 3})$$

$$F1 + \Delta \circ F = (\text{п. 5 табл. 3})$$

$$\varphi \circ F1 + \Delta \circ F$$

$$F1 = \varphi \circ F1 (\text{п. 5 табл. 3}), \text{ ч.т.д.}$$

Утверждение 4. Выражение вида $AFEG\varphi$ логики CTL может быть представлено в RTL-нотации в виде системы рекурсивных уравнений:

$$F = \varphi \circ (F1 + \Delta^\omega) + \Delta \circ F$$

$$F1 = \varphi \circ (F1 + \Delta^\omega)$$

Доказательство. Обозначим через F выражение $AFEG\varphi = AF(EG\varphi)$, а через $F1$ выражение $AE\varphi$.

$$F = AFEG\varphi = AF(EG\varphi) = (\text{п. 3 табл. 3})$$

$$F1 + \Delta \circ F = (\text{п. 6 табл. 3})$$

$$\varphi \circ (F1 + \Delta^\omega) + \Delta \circ F$$

$$F1 = \varphi \circ (F1 + \Delta^\omega) (\text{п. 6 табл. 3}), \text{ ч.т.д.}$$

Докажем также теорему, существенно упрощающую формулировку некоторых верифицируемых условий.

Теорема 2. Справедливы следующие равенства:

1. $AFAF\varphi = AF\varphi$;
2. $EFEF\varphi = EF\varphi$;
3. $AGAG\varphi = AG\varphi$;
4. $EGEG\varphi = EG\varphi$.

Доказательство п. 1. Обозначим через $F1$ выражение $AFAF\varphi$, а через $F2$ выражение $AF\varphi$.

$$F1 = AFAF\varphi = F2 + \Delta \circ F1 = \varphi + \Delta \circ F2 + \Delta \circ F1 =$$

$$= (A13) = \varphi + \Delta \circ (F2 + F1) = \varphi + \Delta \circ F3$$

$$F3 = (F2 + F1) = \varphi + \Delta \circ F2 + \Delta \circ (F2 + F1) = (A13) = \varphi + \Delta \circ (F2 + F2 + F1) = \varphi + \Delta \circ (F2 + F1) = \varphi + \Delta \circ F3.$$

Таким образом, для левой части равенства получаем систему из двух рекурсивных уравнений:

$$F1 = \varphi + \Delta \circ F3$$

$$F3 = \varphi + \Delta \circ F3.$$

Очевидно, что рекурсивное уравнение, представляющее выражение $AF\varphi$, можно расширить до системы уравнений вида

$$F1' = \varphi + \Delta \circ F2'$$

$$F2' = \varphi + \Delta \circ F2'.$$

Получаем систему уравнений, совпадающую с системой уравнений для левой части равенства 1 с точностью до переименования метапеременных, а следо-

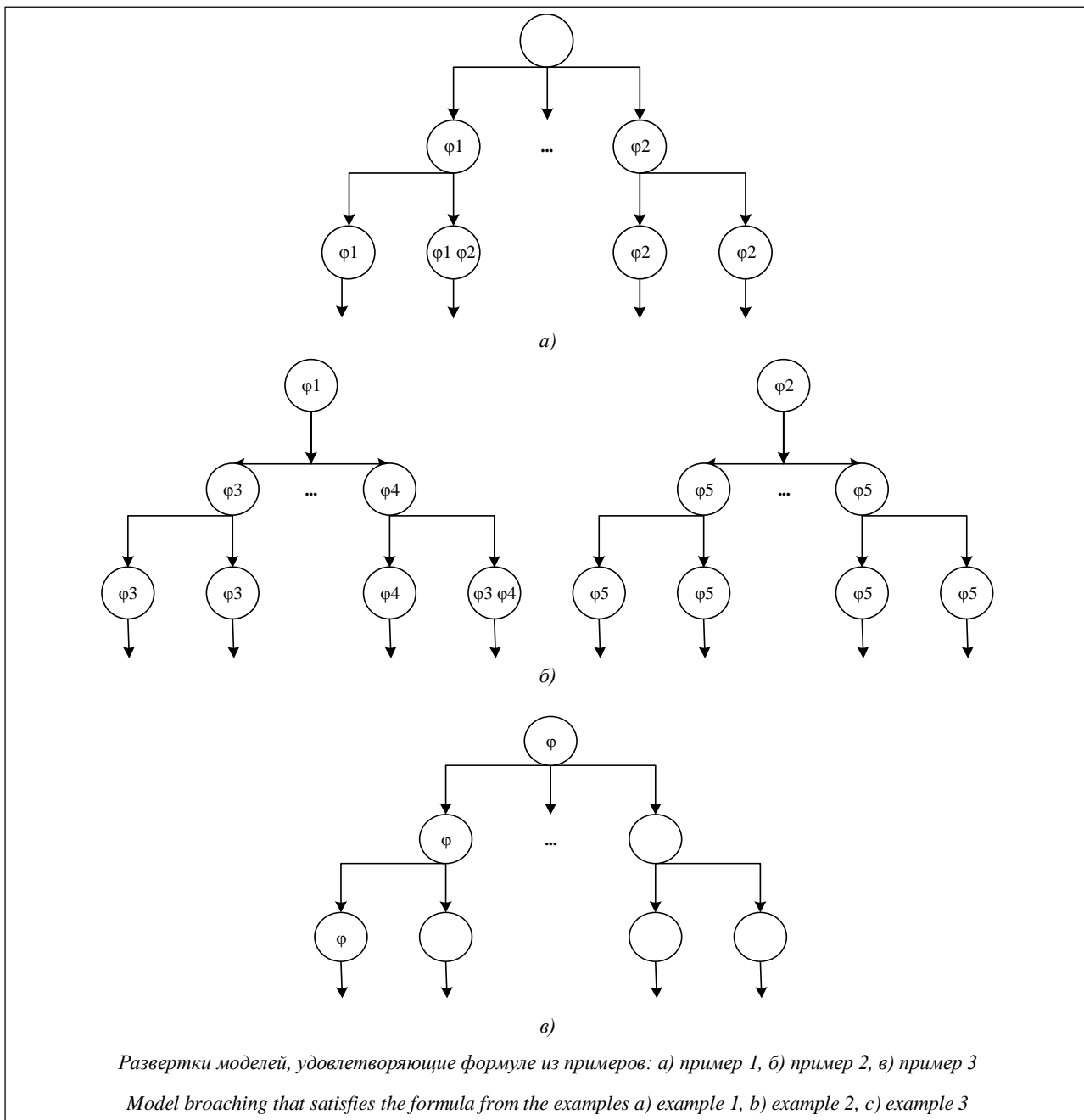
вательно, задающую то же самое множество последовательностей, ч.т.д.

Аналогичным образом нетрудно доказать справедливость равенств 2–4.

Примеры

Рассмотрим три примера для лучшего понимания принципов задания формул ветвящегося времени на базе RTL.

1. $F = \Delta \circ (G\varphi_1 + G\varphi_2)$. Формула истинна, если «из текущего состояния исходят два типа ветвей: либо во всех состояниях выполняется φ_1 , либо во всех состояниях выполняется φ_2 ».



2. $F = \varphi_1 \circ (G\varphi_3 + G\varphi_4) + \varphi_2 \circ G\varphi_5$. Формула истинна, если «в текущем состоянии выполняется φ_1 или φ_2 , при этом, если выполняется φ_1 , то из текущего состояния исходят два типа ветвей: либо во всех состояниях выполняется φ_3 , либо во всех состояниях выполняется φ_4 ; если в текущем состоянии выполняется φ_2 , то исходят лишь те ветви вычислительных последовательностей, для которых во всех состояниях истинно φ_5 ».

3. $F = \varphi \circ (F + \Delta^\omega)$. Формула истинна, если «в текущем состоянии выполняется φ и из него исходит хотя бы одна ветвь, в которой будет истинна эта же формула».

На рисунке представлены развертки моделей, удовлетворяющие примерам 1–3 соответственно.

Отметим, что переход в некоторое состояние, помеченное каким-либо символом предиката (например, предикатом φ), отображает на самом деле все множество переходов в состояния, в которых выполняется этот предикат.

Заключение

В статье была рассмотрена RTL-нотация, представляющая собой расширенную и адаптированную под логику ветвящегося времени RLTL-нотацию. Наглядно на конкретных примерах продемонстрировано, что RTL-нотация способна не только в полной мере выражать базовые конструкции CTL, позволяя формулировать требования к альтернативным путям развития вычислительного процесса верифицируемой модели, но и расширять выразительную мощность логики, выражая утверждения, которые не могут быть выражены в CTL. Кроме того, представленная верифицируемых условий с помощью RTL-нотации позволяет упростить процесс построения композиции условий и верифицируемых моделей. Вопросы решения

задачи Model Checking и ее сложности с помощью RTL-нотации будут подробно рассмотрены в последующих работах.

Таким образом, RTL-нотация в целом представляет собой гибкий и мощный механизм, расширяющий выразительность обеих логик (LTL и CTL) и делающий процесс верификации более эффективным и удобным. В силу своих особенностей RTL может стать единым универсальным инструментом для выполнения всесторонних проверок свойств моделей, позволив специалистам по верификации избавиться от необходимости выполнять лишние действия, связанные с использованием большого числа различных структур данных и подходов к верификации.

Литература

1. Карпов Ю.Г. Model Checking. Верификация параллельных и распределенных программных систем. СПб: БХВ-Петербург, 2010. 552 с.
2. Кларк Э.М., Грамберг О., Пелед Д. Верификация моделей программ. Model Checking. М.: МЦНМО, 2002. 416 с.
3. Pnueli A. The temporal logic of program. Proc. 18th Anny. Symp. on Foundation of Computer Science, 1977, pp. 46–57.
4. Manna Z., Pnueli A. The temporal logic of reactive and concurrent systems. Specification, 1992, 427 p.
5. Kroger F., Merz S. Temporal logic and state systems. Springer, 2008, 436 p.
6. Huth M., Ryan M. Logic in computer science: modelling and reasoning about systems. Cambridge Univ. Press, 2004, 443 p.
7. Кораблин Ю.П., Шипов А.А. Эквивалентная характеристика формул LTL // Программные продукты и системы. 2015. № 4. С. 175–179.
8. Кораблин Ю.П., Шипов А.А. Построение моделей систем на базе эквивалентной характеристики формул LTL // Программные продукты и системы. 2017. № 1. С. 61–66.
9. Olderog E.-R., Apt K.R. Fairness in parallel programs: the transformational approach. ACM Transactions on Programming Languages and Systems, 1988, vol. 10, no. 3, pp. 420–455.
10. Yongming Li, Yali Li, Zhanyou Ma. Computation tree logic model checking based on possibility measures. Fuzzy Sets and Systems, 2015, vol. 262, pp. 44–59.

The unified representation of LTL and CTL logics formulas by recursive equation systems

Yu.P. Korablin^{1,2}, Dr.Sc. (Engineering), Professor, y.p.k@mail.ru
A.A. Shipov³, Ph.D. (Engineering), Senior Engineer-Programmer, a-j-a-1@yandex.ru

¹Russian Technological University (MIREA), Moscow, 119454, Russian Federation

²National Research University "Moscow Power Engineering Institute", Moscow, 111250, Russian Federation

³RSC Group, Moscow, 121170, Russian Federation

Abstract. Nowadays, to solve the formal verification problem using the Model Checking method, the following logics are often used: the linear-time temporal logic (LTL), the computation tree logic (CTL) and CTL* that combines the capabilities of both other logics. However, each of these logics has its own disadvantages, limitations and expressiveness problems due to their syntactic and semantic features. Therefore, there is no universal temporal logic at the moment.

The authors are convinced that special representations, which are based on systems of recursive equations in regard to temporal logics, can extend their expressiveness, as well as unify their syntax. Thus, they allow building their common and uniform notation.

The paper proposes and considers a special RTL notation that is based on systems of recursive equations and the accustomed LTL and CTL semantic definitions. The notation is intended to solve the problem of unification of expressiveness of both logics, which in

turn expands the expressiveness each one of them. The unification of their syntactic structures will give an opportunity to develop a uniform approach for the Model Checking problem.

The authors provide a detailed definition of the RTL notation; give corresponding axioms and theorems. The paper also represents a number of examples and statements that clearly demonstrate the RTL expressiveness capabilities.

The purpose of the paper is to demonstrate key features and capabilities of the RTL notation, which are the basis for the authors' further research on solving the problem of system models verification.

Keywords: verification, Model Checking, RLTL equational characteristics, temporal logic formula, LTL, CTL.

References

1. Karpov Yu.G. *Model Checking. Verification of Parallel and Distributed Software Systems*. St. Petersburg, BHV-Peterburg Publ., 2010, 552 p.
2. Clarke E.M., Grumberg O., Peled D.A. *Model Checking*. MIT Press, Cambridge, MA, USA, 1999 (Russ. ed.: Moscow, MTsNMO Publ., 2002, 416 p.).
3. Pnueli A. The temporal logic of program. *Proc. 18th Annyv. Symp. on Foundation of Computer Science*. 1977, pp. 46–57.
4. Manna Z., Pnueli A. *The Temporal Logic of Reactive and Concurrent Systems*. Specification, 1992, 427 p.
5. Kroger F., Merz S. *Temporal Logic and State Systems*. Springer Publ., 2008, 436 p.
6. Huth M., Ryan M. *Logic in Computer Science: Modelling and Reasoning About Systems*. Cambridge Univ. Press, 2004, 443 p.
7. Korablin Yu.P., Shipov A.A. LTL formula equational characteristics. *Software & Systems*. 2015, no. 4, pp. 175–179 (in Russ.).
8. Korablin Yu.P., Shipov A.A. System models construction based on LTL formula equational characteristics. *Software & Systems*. 2017, no. 1, pp. 61–66 (in Russ.).
9. Olderog E.-R., Apt K.R. Fairness in parallel programs: The transformational approach. *ACM Trans. on Programming Languages and Systems*. 1988, vol. 10, no. 3, pp. 420–455.
10. Li Y., Li Y., Ma Zh. Computation tree logic model checking based on possibility measures. *Fuzzy Sets and Systems*. 2015, vol. 262, no. 1, pp. 44–59.

Примеры библиографического описания статьи

1. Кораблин Ю.П., Шипов А.А. Унифицированное представление формул логик LTL и CTL системами рекурсивных уравнений // Программные продукты и системы. 2019. Т. 32. № 1. С. 20–25. DOI: 10.15827/0236-235X.125.020-025.
2. Korablin Yu.P., Shipov A.A. The unified representation of LTL and CTL logics formulas by recursive equation systems. *Software & Systems*. 2019, vol. 32, no. 1, pp. 20–25 (in Russ.). DOI: 10.15827/0236-235X.125.020-025.