

©Вега Висе Х. Л., Михайлов В. Ю., 2018

DOI: 10.18255/1818-1015-2018-5-481-490

УДК 517.9

О методах верификации и разработки программ развития сельскохозяйственных территорий

Вега Висе Х. Л., Михайлов В. Ю.

получена 20 июля 2018

Аннотация. В настоящее время повсеместными стали методы программно-целевого управления развитием различных социально-экономических систем сложной структуры, например, таких как территории сельскохозяйственного назначения. Поэтому актуальными задачами являются верификация уже созданных программ развития и разработка «правильных» программ развития таких систем, по аналогии с верификацией и разработкой правильных компьютерных программ – развитыми дисциплинами в теоретическом программировании. В данной работе для решения задачи верификации программ развития сельскохозяйственных территорий сначала строится структурная схема программы, по которой создается аксиоматическая теория, использующая аппарат алгоритмических логик Хоара. Основной проблемой при построении аксиоматической теории является разработка аксиом теории, отражающих предусловия и эффекты выполнения содержательных действий, указанных в тексте программы развития. Верификация программы развития соответствует проверке доказуемости некоторой тройки Хоара, соответствующей начальным и целевым условиям программы. Для задачи разработки правильных программ развития описывается механизм построения модели предметной области с использованием языков описания моделей семейства PDDL. Описание конкретной модели имеет чисто декларативный характер и представляет собой набор описаний предикатов и действий выбранной предметной области. Показывается, как на описанной модели с помощью интеллектуальных планировщиков, включая темпоральные планировщики типа ОРТИС, автоматически строить решения целевых задач программ развития. На основе экспертных знаний и отраслевых стандартов построена модель сельскохозяйственной территории, краткое описание которой приводится в работе. Проведенные эксперименты показали эффективность предлагаемого подхода к разработке правильных программ развития.

Ключевые слова: программы развития, верификация программ развития, разработка правильных программ развития, логики Хоара, язык PDDL, интеллектуальные планировщики, модель сельскохозяйственной территории

Для цитирования: Вега Висе Х. Л., Михайлов В. Ю., "О методах верификации и разработки программ развития сельскохозяйственных территорий", *Моделирование и анализ информационных систем*, 25:5 (2018), 481–490.

Об авторах:

Вега Висе Хорхе Луис, orcid.org/0000-0003-4323-722X, аспирант,
Казанский (Приволжский) федеральный университет,
ул. Кремлёвская, 18, г. Казань, 420008 Россия, e-mail: jvegavice@gmail.com

Михайлов Валерий Юрьевич, orcid.org/0000-0002-2778-1797, канд. физ.-мат. наук, доцент,
Казанский (Приволжский) федеральный университет,
ул. Кремлёвская, 18, г. Казань, 420008 Россия, e-mail: Valery.Mikhailov@kpfu.ru

Введение

Среди многочисленных значений слова «программа» в толковых словарях русского языка выделяются три: (1) краткое содержание работы или деятельности (например, программа передач радио или телевидения, программа концерта и т.п.); (2) план деятельности, совокупность действий и мероприятий для осуществления поставленных целей, формулировка которых также часто является частью программы (например, программа партии, программа «500 дней», программа учебной дисциплины); (3) программа для компьютера, которая представляет собой запись на некотором формальном языке алгоритма решения определенного множества задач.

Программы развития (ПР) территорий, естественно, не являются программами для компьютеров, но разрабатываются столь подробно, что значение слова «программа» в них содержит явные признаки как 2-го, так и 3-го значений из вышеуказанного списка. Поэтому представляется естественным применять к их верификации и разработке соответствующие методы, развитые в теоретическом программировании для компьютерных программ.

В данной работе предлагаются подход к применению проверки доказуемости теорем в одной из *алгоритмических логик Хоара* [1] для верификации ПР территорий, а также описание модели предметной области на языке PDDL (Planning Domain Definition Language) [2] и использование программ *темпоральных планировщиков* [3] для получения последовательности действий, позволяющей достичь желаемого состояния территории, на основе которой легко строится необходимая правильная ПР.

1. Верификация ПР

Любая ПР рассматривает территорию как объект управления, который необходимо перевести из данного начального состояния S_0 в некоторое целевое состояние S_k за k периодов времени (месяцев, кварталов, лет). Состояние территории моделируется конечным списком параметров $P = \langle p_1, p_2, \dots, p_n \rangle$, например, таких как «содержание фосфора в почве», «содержание азота в почве», «влажность почвы», «число обрабатывающих механизмов», «объем имеющихся финансовых ресурсов на развитие» и т.п. Выделяются состояния территории, относящиеся к концу определенного временного периода. Состояние территории S_t в конце периода t в ПР описывается набором действительных чисел $P_t = \langle p_{t1}, p_{t2}, \dots, p_{tn} \rangle$, где $p_{t1}, p_{t2}, \dots, p_{tn}$ – конкретные значения параметров p_1, p_2, \dots, p_n соответственно.

В каждой ПР указывается некоторое множество базовых действий $E = \{e_1, \dots, e_s\}$, часто называемые мероприятиями, выполнение которых в каждый период времени должно приводить к изменению состояния территории. Выполнения действия e_i в разные периоды времени могут отличаться друг от друга интенсивностью, которая зависит, например, от выделенных финансовых средств на выполнение данного действия в тот или иной период времени. Оператор $e_i(a_1, a_2, \dots, a_k)$ обозначает выполнение действия e_i с атрибутами a_1, a_2, \dots, a_k . В ПР мероприятия в любой период времени могут выполняться либо последовательно, либо параллельно друг другу.

Введем понятие **выполнимой программы**.

1) Любой оператор $e_i(a_1, a_2, \dots, a_k)$ является выполнимой программой.

2) Если π_1 и π_2 – выполнимые программы, то $(\pi_1; \pi_2)$ и $(\pi_1 \parallel \pi_2)$ также являются выполнимыми программами.

Содержательно $(\pi_1; \pi_2)$ означает последовательное выполнение программ π_1 и π_2 , а $(\pi_1 \parallel \pi_2)$ – параллельное выполнение программ π_1 и π_2 . Например, выполнимая программа $((e_1(a_1) \parallel e_2(a_2)); (e_3(a_3) \parallel e_4(a_4)))$ соответствует параллельному выполнению операторов $e_1(a_1)$ и $e_2(a_2)$, за которым параллельно выполняются операторы $e_3(a_3)$ и $e_4(a_4)$.

Обзор многочисленных программ развития территорий показал, что структуру каждой ПР территории можно представить в виде схемы S:

$$P_0 \rightarrow \pi(1) \rightarrow P_1 \rightarrow \pi(2) \rightarrow \dots \rightarrow P_{t-1} \rightarrow \pi(t) \rightarrow P_t \rightarrow \dots \rightarrow P_{k-1} \rightarrow \pi(k) \rightarrow P_k,$$

где часть $P_{t-1} \rightarrow \pi(t) \rightarrow P_t$ соответствует структуре временного периода t , $t = 1, \dots, k$, P_{t-1} – состояние территории в начале периода t , P_t – состояние территории в конце периода t , $\pi(k)$ – выполнимая программа.

Под спецификацией состояния P_t будем понимать его описание ϕ_t на некотором формализованном языке L.

Пусть ϕ и ψ – формулы языка L, а π – выполнимая программа.

Будем говорить, что справедливо утверждение $\{\phi\}\pi\{\psi\}$, если всякий раз, когда перед выполнением π истинна формула ϕ , после выполнения этой программы будет истинна формула ψ .

Таким образом, проверка правильности ПР территории заключается в проверке справедливости утверждений: $\{\phi_{t-1}\}\pi(t)\{\phi_t\}$, $t=1, \dots, k$. А справедливость утверждений $\{\phi_{t-1}\}\pi(t)\{\phi_t\}$ мы будем понимать как доказуемость этих утверждений в определенной теории T, соответствующей ПР и основанной на определенной алгоритмической логике.

В качестве языка спецификаций для теории T выберем формулы вида

$$p_1 \geq r_1 \& p_2 \geq r_2 \& \dots \& p_n \geq r_n, \quad (1)$$

где p_1, \dots, p_n – переменные, соответствующие параметрам, а r_1, r_2, \dots, r_n – некоторые арифметические выражения.

Правильно построенными выражениями теории T являются выражения вида $\{\phi\}\pi\{\psi\}$, где формулы ϕ и ψ имеют вид (1).

Аксиомами теории T являются выражения вида

$$\{p_{i1} \geq x_1 \& \dots \& p_{im} \geq x_m\} e(a_1, \dots, a_k) \\ \{p_{i1} \geq x_1 + g_1(a_1, \dots, a_k) \& \dots \& p_{im} \geq x_m + g_m(a_1, \dots, a_k)\}, \quad (2)$$

где p_{i1}, \dots, p_{im} – переменные, соответствующие параметрам, x_1, \dots, x_m – произвольные значения параметров p_{i1}, \dots, p_{im} , $g_j(a_1, \dots, a_k)$ – изменение параметра p_{ij} после выполнения оператора $e(a_1, \dots, a_k)$, $j=1, \dots, m$. Заметим, что если все атрибуты оператора $e(a_1, \dots, a_k)$ заданы, то каждое $g_j(a_1, \dots, a_k)$ есть некоторое действительное число (положительное или отрицательное).

Аксиомы линейного порядка также являются аксиомами теории T.

Правила вывода теории T.

$$\text{П1 : } \{\phi\}\pi\{\psi\}, (\psi \rightarrow \gamma) \vdash \{\phi\}\pi\{\gamma\} \\ \text{П2 : } \{\phi\}\pi\{\psi\}, (\gamma \rightarrow \phi) \vdash \{\gamma\}\pi\{\psi\}$$

$$\begin{aligned} \text{ПЗ} : \{ \phi \} \pi_1 \{ \gamma \}, \{ \gamma \} \pi_2 \{ \psi \} \vdash \{ \phi \} (\pi_1; \pi_2) \{ \psi \} \\ \text{П4} : \{ \phi \} \pi_1 \{ \psi_1 \}, \{ \phi \} \pi_2 \{ \psi_2 \} \vdash \{ \phi \} (\pi_1 \parallel \pi_2) \{ \psi_1 \&_{\phi}^+ \psi_2 \} \end{aligned}$$

Здесь символ « \rightarrow » означает импликацию, символ « \vdash » - логическую выводимость, символ « $\&_{\phi}^+$ » - операцию над формулами вида (1), вычисляемую по правилу: если $p_i \geq a_i$ входит в ϕ , $p_i \geq b_i$ входит в ψ_1 , $p_i \geq c_i$ входит в ψ_2 , то $p_i \geq a_i + (b_i - a_i) + (c_i - a_i)$ входит в формулу $\psi_1 \&_{\phi}^+ \psi_2$.

Никакие другие подформулы в формулу $\psi_1 \&_{\phi}^+ \psi_2$ не входят.

Заметим, что в схеме S спецификациями состояний P_t являются формулы вида (1), в которых все выражения r_1, r_2, \dots, r_n - действительные числа, и в выполнимых программах $\pi(t)$ атрибуты всех входящих в них операторов заданы. Не трудно показать, что задача определения доказуемости троек $\{ \phi_{t-1} \} \pi(t) \{ \phi_t \}$, $t = 1, \dots, k$, в теории T является эффективно разрешимой.

Отметим, что построение множества аксиом теории T является самым содержательным и трудоемким этапом построения теории T, создаваемой для проверки правильности ПР. Очевидно, что они должны отражать мнения экспертов по развитию территорий, согласовываться с данными статистики и соответствовать определенной методологии обработки ресурсов для устойчивого территориального развития [4].

2. Разработка ПР

Для разработки ПР с/х территории нами предлагается на языке PDDL создавать модель этой территории, состоящую из набора предикатов и действий, соответствующих определенным способам обработки различных ресурсов территориального развития, и использовать для построенной модели алгоритмы «интеллектуального планирования», т.е. алгоритмы поиска последовательностей действий, реализация которых переведет начальное состояние территории в целевое.

2.1. Язык PDDL (Planning Domain Definition Language)

Язык PDDL [5] - язык описания областей и задач планирования, появившийся в ответ на потребность в едином, универсальном для всех планировщиков языке. Основную структуру и синтаксис PDDL унаследовал от одного из своих предшественников - языка STRIPS (Stanford Research Institute Problem Solver) [6], который был создан для решения конкретной задачи управления роботом. Язык PDDL носит описательный характер, не содержит алгоритмов как таковых и предлагает стандарт представления компактных и простых моделей для спецификации областей и задач планирования.

2.2. Описание модели в PDDL

Модель в PDDL всегда представляется в двух файлах: файл домена и файл проблемы. Файл домена содержит описание предметной области в целом: описания типов существующих объектов, описания констант, описания предикатов (свойств) и

описания действий, используемых в данной области. Файл проблемы содержит конкретную постановку задачи планирования: описание конкретных взаимодействующих объектов, начальное и целевое состояния моделируемой системы. Отделение описания домена от описания проблемы позволяет использовать модель одного и того же домена многими пользователями в решении своих задач разработки планов развития.

Файл домена содержит все возможные предикаты и действия, которые нужны для описания системы. Предикаты описываются в стандартной называющей форме, где после названия предиката перечисляются его аргументы с указанием их типов. После описания предикатов в файле домена идёт блок описания действий. Описание каждого из действий состоит из трёх основных частей: параметры действия, предусловие выполнения действия и эффект выполнения действия. Параметры содержат локальные имена переменных с указанием их типов; предусловие содержит выражение из предикатов и логических связок и должно быть истинным, чтобы действие было выполнимо; эффект содержит предикаты, соединённые связкой «и», которые станут истинными после выполнения действия.

Необходимо отметить, что PDDL постоянно эволюционирует, расширяя свои выразительные средства описания моделей. В настоящее время существует довольно большое количество планировщиков, которые способны интерпретировать модели, определенные в PDDL. Среди них можно выделить SGPlan, TFD, LPD, VHPOP, HSP, CTP, POPF, OPTIC, COLIN, FastDownward [7,8,9]. Для наших целей самым удобным и производительным оказался темпоральный планировщик OPTIC [10], разработанный для расширения языка PDDL2.1 [11], на котором мы и производили эксперименты с моделями.

Приведем формат описания простого и темпорального действий в PDDL2.1 соответственно.

```
(:action <имя действия>
:parameters ( <список параметров действия> )
:precondition (and ( <условие> ) ) /* условие, которое должно выполняться
                                при запуске действия
:effect (and ( <предикат> ) ... )
) /* конец действия
(:durative-action <имя действия>
:parameters ( <список параметров действия> )
:duration ( <продолжительность действия> )
:condition (and (at start ( <условие> ) ) /* условие, которое должно выполняться
                                при запуске действия
                (over all ( <условие> ) ) /* условие, которое должно выполняться
                                во время исполнения действия
                (at end ( <условие> ) ) /* условие, которое должно выполняться
                                в конце исполнения действия
                ) /* конец условий
:effect (and (at start ( <предикат> ) ) ...
            (at end ( <предикат> ) )
            ) /* конец эффектов
) /* конец действия
```

Далее, в качестве простого примера приведем фрагмент экспериментальной модели работы сельскохозяйственной фермы.

```
(:action to-applyFertilizer
:parameters (?f - fertilizer ?p - plot)
:precondition (and (it-has THE-FARM ?f) )
:effect (and
      ( when (it-has ?f nutrient-P) (it-has ?p nutrient-P) )
      ( when (it-has ?f nutrient-N) (it-has ?p nutrient-N) )
      ( when (it-has ?f nutrient-K) (it-has ?p nutrient-K) )
      (not (it-has THE-FARM ?f))
    )
) /* конец действия
(:durative-action to-sowPlot
:parameters (?p - plot ?c - crop ?s - seed)
:duration (= ?duration (time-to-sow-crop ?c) )
:condition (and
      (at start (plot-isUnsown ?p) )
      (at start (it-has THE-FARM ?s) )
      (at start (it-for ?s ?c) )
      (at start (it-has ?p plHumidity) )
      (at start (seed-forCrop ?s ?c) )
      (at start (it-has ?p nutrient-P) )
      (at start (it-has ?p nutrient-N) )
      (at start (it-has ?p nutrient-K) )
      (over all (not (it-has ?p plSalinity) ) )
      (over all (not (it-has ?p plHeavyMetals) ) )
      (over all (not (it-has ?p plOil) ) )
    )
:effect (and
      (at end (plot-isSown ?p) )
      (at end (not (plot-isUnsown ?p) ) )
      (at end (plot-isSownWithCrop ?p ?c ?s) )
      (at end (not (it-has THE-FARM ?s)) )
      (at end (increase (total-elapsed-time) ?duration))
    )
) /* конец действия
```

В действии «to-applyFertilizer» используется предикат «(it-has ?x ?y)», который не ограничивает тип его параметров и позволяет утверждать, что «?x имеет ?y». В предусловии этого действия выражение «(and (it-has THE-FARM ?f))» означает, что действие может быть выполнено, только если на объекте «THE-FARM» (ферма) имеется удобрение «?f». В эффектах действия выражение «(when (it-has ?f nutrient-P) (it-has ?p nutrient-P))» означает, что если удобрение «?f» содержит питательное вещество «nutrient-P» (фосфор), то участок «?p» будет содержать это питательное вещество (то же самое с питательными веществами «nutrient-N» (азот) и «nutrient-K» (калий)).

Действие «to-sowPlot» является темпоральным («durative-action») и требует указания продолжительности его выполнения, используя в этом случае простую численную функцию «time-to-sow-crop ?c», которая означает, как долго длится процесс посева культуры «?c». Это действие выполняется, если в начале («at start») участок «?p» не засеян, на ферме есть семена типа «?s», участок имеет необходимую влажность, семена типа «?s» предназначены для культивирования «c» («it-for ?s ?c»), и если почва участка содержит необходимые питательные вещества «P», «N» и «K», но не содержит загрязняющие вещества «plSalinity», «plHeavyMetals» и «plOil» постоянно («over all»). В результате завершения действия («at end») участок «?p» будет засажен семенами типа «?s», а в «THE-FARM» больше не будет семян такого типа. В конце действия значение выражения «(increase (total-elapsed-time) ?duration)», равное продолжительности выполнения действия, добавляется к численному значению функции «total-elapsed-time», которая контролирует общее время выполнения целевой последовательности действий.

Отметим определенное сходство формата описания действий в PDDL и вида аксиом в теории T, строящейся для конкретной программы развития при ее верификации. С помощью описанных конструкций языка PDDL2.1 нам удалось разработать модель развития с/х территорий, кратко описанную ниже.

С точки зрения алгоритмов планирования в стандартном PDDL доступны два вида поиска решения: от начального состояния в целевое и из целевого в начальное. В первом случае, если есть несколько способов достичь цели, будет выдан кратчайший по установленной метрике (если не оговорено – по количеству действий) путь. Второй вид поиска (обратный, backward search) доступен в PDDL в силу структуры самого языка, а именно – его действий. Прямой поиск (forward search) обрабатывает действия в указанном порядке (сначала предусловие, затем применяется эффект), в то время как обратный как бы меняет эффект и предусловие местами.

2.3. Модели развития сельскохозяйственных территорий

Действующая классификация земельного фонда РФ предусматривает выделение следующих категорий пригодности земель [12]: I – земли, пригодные под пашню; II – земли, пригодные преимущественно под сенокосы; III – земли пастбищные, после улучшения могут быть пригодны под другие сельскохозяйственные угодья; IV – земли, пригодные под сельскохозяйственные угодья после коренных мелиораций; V – земли, малоприспособленные под сельскохозяйственные угодья; VI – земли, непригодные под сельскохозяйственные угодья; VII – нарушенные земли. Основанием для выделения категорий пригодности является качественное состояние земель и возможность их использования под основные сельскохозяйственные угодья. В отдельных случаях в зависимости от экономических и других факторов существующее использование земель может не соответствовать их намеченной пригодности. Например, земли, пригодные под пашню, если они расположены вблизи населённых пунктов или животноводческих комплексов, могут использоваться для посадки многолетних плодовых насаждений или создания долголетних культурных сенокосов и пастбищ.

Мы кратко охарактеризуем модель развития территорий первых трех типов. К разработке адекватной модели привлекались эксперты по сельскому хозяйству и

территориальному развитию и учитывались отраслевые стандарты министерства сельского хозяйства РФ ОСТ 10 294-2002 - ОСТ 10 297-2002. Ниже будем считать, что исследуемая нами с/х территория состоит из отдельных участков.

Опишем содержание *файла домена* модели «С/х территория».

Параметры и предикаты:

1) Экологические факторы участка, необходимые для нормального роста и развития растений, формирования урожая и его качества, недопущения деградации земель (закисление, засоление, переуплотнение, эрозия, дефляция, истощение запасов органического вещества и доступных для растений питательных элементов, загрязнение вредными веществами и т.д.).

2) Факторы участка, удовлетворяющие потребности сельскохозяйственных культур с учетом их биологических особенностей в питательных элементах (N, P, K, Ca, Mg, S, микроэлементы), воде, воздухе, тепле, фитосанитарных и эколого-токсикологических условий.

3) Высокопродуктивные, адаптированные к местным условиям сорта (гибриды) с/х культур.

Действия:

Группа действий 1: Осуществление агротехнических, агрохимических, мелиоративных, фитосанитарных, противоэрозионных и иных мероприятий по обеспечению плодородия почв;

Предусловия: определенные потребности с/х участка в минеральных удобрениях, химических мелиорантах, пестицидах и других агрохимикатах, органических удобрениях, оборудовании и машинах.

Эффекты: повышение плодородия почвы.

Группа действий 2: Мероприятия по реабилитации земель сельскохозяйственного назначения, загрязненных тяжелыми металлами, пестицидами, радиоактивными веществами и другими токсикантами;

Предусловия: загрязненность с/х участка, наличие соответствующего оборудования и машин;

Эффекты: нормализация экологического состояния почвы.

Группа действий 3: Мероприятия по оптимальному размещению посевов с/х культур, формированию севооборотных массивов, выявлению малопродуктивных земель, трансформации пашни в менее интенсивные виды угодий.

Группа действий 4: Технологические действия возделывания сельскохозяйственных культур с учетом складывающихся погодных и хозяйственно-экономических условий, минерального питания растений, фитосанитарного обследования посевов, запасов продуктивной влаги и плотности почвы в период вегетации растений.

Эффекты: повышение качества и урожайности культур, сокращение затрат на их производство, предотвращение загрязнения окружающей среды средствами химизации.

Описание *файла проблемы* модели «С/х территория».

Здесь описывается набор земельных участков, составляющих территорию, и их

начальные характеристики, включающие состав почв, близость рынков, близость источников чистой воды и др.

Кроме этого, в этом файле указываются целевые показатели развития территории:

1) Улучшение качества участков.

Здесь качество участка – это комплексная характеристика участка по уровню его плодородия и производительной способности, вычисляемая на основе стандартных характеристик земли (гранулометрический состав, эрозия, засоление, избыточное увлажнение, каменистость и т.д.).

2) Планируемая экономическая выгода с/х производства на данной территории.

Наши эксперименты с построенной моделью «С/х территория» и использованием планировщика ОПТИС, разработанного в King’s College London, показали успешность применения описанного подхода к созданию ПР как для с/х компаний, так и для специфических территориально-экономических кластеров.

Список литературы / References

- [1] Hoare C.A.R., “An Axiomatic Basis for Computer Programming”, *Communications of the ACM*, **12**:10 (1969), 576–580.
- [2] McDermott D.V., “PDDL—The Planning Domain Definition Language”, *Tech. Rep. TR-98-003/DCS TR-1165*, Yale Center for Computational Vision and Control, 1998.
- [3] Coles A.J. et al., “Temporal Planning in Domains with Linear Processes”, *Proceedings of the 21st International Joint Conference on Artificial Intelligence*, 2009, 1671–1676.
- [4] Vargas Rodriguez H. et al., “Metodología para el uso y manejo social del recurso tierra como contribución al desarrollo local sostenible”, *Revista de Gestión del conocimiento y el desarrollo local – CEDAR/UNAH*, **1** (2014), 34–38; [Vargas Rodriguez H. et al., “Methodology for the use and social handling of the resource land as a contribution to the sustainable local development”, *Knowledge management and local development magazine – CEDAR/UNAH*, **1** (2014), 34–38, (in Spanish).]
- [5] Helmert M., *An Introduction to PDDL*, Tech. rep., <https://www.cs.toronto.edu/~sheila/2542/s14/A1/introtopddl2.pdf>.
- [6] Fikes R., Nilsson N., “STRIPS: a new approach to the application of theorem proving to problem solving”, *Artificial Intelligence*, **2** (1971), 189–208.
- [7] Gerevini A., Long D., *Plan constraints and preferences in PDDL3*, Tech. Rep., Dept. of Electronics for Automation, University of Brescia, Italy, 2005.
- [8] Cresswell S., Coddington A., “Compilation of LTL goal formulas into PDDL”, *Proceedings of the 16-th European Conference on Artificial Intelligence*, 2004, 985–986.
- [9] Coles A.J. et al., “COLIN: Planning with Continuous Linear Numeric Change”, *Journal of Artificial Intelligence Research*, **44**:1 (2012), 1–96.
- [10] *OPTIC: Optimising Preferences and Time-Dependent Costs*, <https://nms.kcl.ac.uk/planning/software/optic.html>.
- [11] Fox M., Long D., “PDDL2.1 : An Extension to PDDL for Expressing Temporal Planning Domains”, *Journal of Artificial Intelligence Research*, **20**:1 (2003), 61–124.
- [12] *Категории пригодности земель*, http://geolike.ru/page/gl_3219.htm; [*Kategorii prigodnosti zemel*, http://geolike.ru/page/gl_3219.htm, (in Russian).]

Vega Vice J. L., Mikhailov V. Y., "On Methods in the Verification and Elaboration of Development Programs for Agricultural Territories", *Modeling and Analysis of Information Systems*, **25:5** (2018), 481–490.

DOI: 10.18255/1818-1015-2018-5-481-490

Abstract. Nowadays, the methods of program-targeted management for the development of various socio-economic systems of complex structure, such as agricultural areas, have become universal. Therefore, the current tasks at hand are the verification of already created development programs and the development of "proper" programs for the development of such systems, by analogy with the verification and development of proper computer programs through developed disciplines in theoretical programming. In this paper, in order to solve the problem of verification of development programs for agricultural territories, a structural scheme of the program is first constructed, through which the axiomatic theory is created, using Hoare's algorithmic logic system. The main problem in the construction of the axiomatic theory is the development of the axioms of the theory reflecting the preconditions and effects of the implementation of meaningful actions indicated in the text of the development program. The verification of the development program corresponds to the probability of some Hoare triplet, according to the initial and target conditions of the program. For the task of elaboration of the right development programs, the mechanism for constructing a domain model using the PDDL family description languages is described. The description of a specific model is purely declarative in nature and consists of descriptions of predicates and actions of the chosen subject area. In this paper, it is shown how on the described model with the help of intelligent planners, including temporal planners such as OPTIC, solutions to the targets of development programs can be automatically built. Based on expert knowledge and activity standards, a model of an agricultural territory is constructed, a brief description of which is given in the work. The conducted experiments showed the effectiveness of the proposed approach for the development of proper development programs.

Keywords: development programs, verification of development programs, development of proper development programs, Hoare's logic, PDDL language, intelligent planners, model of agricultural territories

On the authors:

Jorge L. Vega Vice, orcid.org/0000-0003-4323-722X, graduate student,
Kazan (Volga region) Federal University,
18 Kremlyovskaya str., Kazan 420008, Russia, e-mail: jvegavice@gmail.com

Valery Y. Mikhailov, orcid.org/0000-0002-2778-1797, PhD,
Kazan (Volga region) Federal University,
18 Kremlyovskaya str., Kazan 420008, Russia, e-mail: Valery.Mikhailov@kpfu.ru