

Минимальная логика для анализа технологий

В. И. Шалак¹

Технологиями могут быть названы алгоритмы, результат выполнения которых является результатом протекания последовательности внешних физических процессов, инициируемых активным агентом. В докладе определен язык и семантика технологий, а также аксиоматизирована минимальная логика технологий. Одной из возможных областей применения предложенного языка является эволюционное программирование с целью адаптивной оптимизации существующих технологий и поиска новых.

Ключевые слова: технологии, логика технологий, натуральные алгоритмы, эволюционное программирование.

1. Введение

Общее неформальное определение понятия алгоритма может быть сформулировано следующим образом:

Алгоритм – это общепонятное и однозначное предписание, какие и в каком порядке производить действия, чтобы получить искомый результат.

Под него подпадают не только символьные вычисления машин Тьюринга, но и более широкий класс явлений, выходящих за рамки операций с символами и существенным образом связанных с оперированием физическими предметами и управлением внешними естественными природными процессами для достижения конкретных целей. Эти явления я буду называть *технологиями* или *натуральными алгоритмами*.

Простейший пример: «Если в комнате темно, включи свет». Все, что могут непосредственно делать люди, – это механически перемещать в различных направлениях свое тело или конечности. В их арсенале нет действия “включить свет”, но они могут *подойти к выключателю и нажать* на него, чтобы *зажечь лампочку*.

Поэтому в развернутом виде предписание могло бы выглядеть следующим образом: «Если в комнате темно, *подойди к выключателю и нажми на кнопку, чтобы тем самым замкнуть электрическую цепь,*

¹ Шалак Владимир Иванович — доктор философских наук, ведущий научный сотрудник Института философии РАН, e-mail: shalack@mail.ru.

Shalack Vladimir Ivanovich — DSc in Philosophy, Leading Research Fellow of Institute of Philosophy, Russian Academy of Sciences.

по которой потечет электрический ток и нагреет спираль лампочки до состояния свечения, чтобы она осветила комнату».

Данный натуральный алгоритм характеризуется следующими чертами:

- 1) Эмпирическое знание C о положении дел – в комнате темно.
- 2) Конкретные манипуляции d , которые должен произвести агент – подойти к выключателю и нажать на кнопку.
- 3) Физический процесс протекания электрического тока и нагревания спирали лампочки P . Манипуляции агента всего лишь запускают внешний физический процесс, который далее протекает без активного участия агента, но является необходимой составляющей алгоритма.
- 4) И наконец, цель выполнения алгоритма G – освещение комнаты.
- 5) Натуральный алгоритм (технология) реализуется не в абстрактном математическом, а в реальном времени в окружении реальных природных процессов.
- 6) Манипуляции d активного агента связывают в единые цепочки различные физические процессы P , которые без его участия никогда бы не произошли в этой очередности и не привели к требуемому результату G .
- 7) Элементарный кирпичик натуральных алгоритмов может быть описан в виде правила «Если имеет место C , произведи манипуляции d , чтобы запустить процесс P для достижения искомой цели G », или кратко " $C \rightarrow d:P:G$ ".

Сложный натуральный алгоритм можно представить, как набор элементарных правил.

Очевидно, что логика для описания и анализа технологий должна совмещать в себе черты временной и динамической логик.

2. Основной результат

Для простоты был взят язык пропозициональной логики, расширенный множеством элементарных действий *Act*, которые может непосредственно произвести агент, а также двумя двухместными временными операторами *Since* и *Until*, которые помимо задания временных отношений могут быть использованы для представления ограниченных во времени

процессов. Эти операторы можно по аналогии с ограниченными кванторами понимать, как ограниченные временные операторы H и G . Если HA интерпретируется, как "всегда было A ", то $S(B, A)$ понимается, как "с некоторого момента в прошлом, когда было B , всегда имело место A ". Аналогично для будущего $U(B, A)$ — "до некоторого момента в будущем, когда станет истинно B , всегда будет иметь место A ".

Так как нас интересует минимальная временная логика, в качестве модельной была взята структура временной логики Kt . Интерпретация элементарных действий, поскольку внешнее физическое время обладает приоритетом, согласована с отношением временного порядка.

Построена полная аксиоматизация логики, в языке которой можно определить понятие выполнения элементарной технологии.

$\langle C \rightarrow d : P : G \rangle A =_{def} C \wedge \langle d \rangle U(A, P)$ — «В результате некоторого выполнения технологии " $C \rightarrow d : P : G$ " имеет место A ».

Поскольку технологии реализуются в реальном мире в окружении других в общем случае неподконтрольных нам процессов, не каждая попытка их выполнения приводит к желаемой цели. Утверждение об успешном выполнении технологии " $C \rightarrow d : P : G$ " может быть представлено следующим образом:

$! \langle C \rightarrow d : P : G \rangle A =_{def} C \wedge \langle d \rangle U(G \wedge A, P)$ — «В результате некоторого успешного выполнения технологии " $C \rightarrow d : P : G$ " имеет место A ».

Предложенный язык и логика могут быть взяты в качестве минимальных для описания и анализа технологий, так как не налагают никаких ограничений на характер временной упорядоченности событий внешнего мира. В будущем в зависимости от области приложения можно рассмотреть варианты линейного и древовидного порядков, дискретного, плотного или непрерывного времени. В последнем случае мы получим язык и логику для совместного описания и анализа, например, дискретных управляющих действий и непрерывных процессов, что имеет отношение к аналоговым алгоритмам.

3. Возможные приложения в ИИ

Эволюционное (генетическое) программирование — направление исследований в ИИ специальных алгоритмов самообучения, опирающихся на известные генетические механизмы изменчивости и наследования полезных изменений. Эти алгоритмы позволяют адаптивным образом из начальных заготовок программ получать более эффективные программы

для решения конкретных задач. Сложная технология может быть представлена в виде набора простых по образу и подобию продукционного программирования. Представление сложных технологий в виде набора элементарных " $C \rightarrow d:P:G$ " хорошо подходит для приложения к ним методов эволюционного программирования, поскольку структурно отделяет эмпирическую информацию C от действий активного агента d , инициируемых им процессов P и преследуемых целей G , что позволяет реализовать механизмы изменчивости и наследования отдельно по каждой из этих компонент. Обратная связь со внешней средой реализуется посредством оценки успешности или неуспешности выполнения технологий.

Натуральные алгоритмы в данном представлении позволяют описывать, анализировать и моделировать на компьютерах не только инженерные технологии, но и различные закономерности функционирования социальной сферы. Благодаря возможности применения алгоритмов эволюционного программирования может быть поставлена задача автоматизации поиска и создания новых более эффективных технологий и моделей поведения.

Minimal logic for technology analysis **Shalack V.I.**

Technologies are algorithms, the result of which is the result of a sequence of external physical processes initiated by an active agent. We define the language and semantics of technologies. Also the minimal logic of technologies is axiomatized. One of the possible areas of application of the proposed language is evolutionary programming for adaptive optimization of existing technologies and the search for new ones.

Keywords: technologies, logic of technologies, natural algorithms, evolutionary programming.