

УДК 378.147

## ВОЗДЕЙСТВИЕ ИМИТАЦИОННЫХ МОДЕЛЕЙ НА ПОЛЕ ВЫЧИСЛИТЕЛЬНОЙ ДИДАКТИКИ

Патаракин Е.Д. (Москва)

### Введение

В настоящее время наблюдается очередной всплеск вычислительного мышления, взаимодействия человека и компьютера. Вычислительное мышление рассматривается многими исследователями как важнейший навык современного человека. Для формирования вычислительного мышления постоянно создаются новые учебные языки программирования, обучающие среды, видеоигры и сетевые сообщества, проектируются средства и сценарии деятельности, направленные на освоение учащимися умений вычислительного мышления. Впервые предложили использовать прилагательное «вычислительный» по отношению к области знаний в своей книге о перцептронах и вычислительной геометрии Пейперт и Минский в 1969 году [1]. В эти же годы появляется язык Logo – ставший одновременно и символом нового философского подхода в образовании, и важной точкой в развитии языков программирования, и родоначальником целой группы специализированных языков многоагентного имитационного моделирования. Усиление возможностей MediaWiki за счёт расширений Semantic Mediawiki, EmbedScratch, ScratchBlocks4, Snap! Project Embed, graphviz, Widgets:iframe, Widgets:YouTube позволяет собирать на одном поле проекты, созданные в различных сетевых образовательных сообществах. Это позволяет учащимся продвигаться от более простых к более сложным примерам реализации решения сходных проблем в различных средах многоагентного программирования. Адрес проекта – <http://digida.mgpu.ru>

### Материалы и методы

Данная работа продолжает тему внедрения в образование сред многоагентного программирования, внутри которых агенты могут взаимодействовать друг с другом и с окружающим миром. Одним из результатов предыдущих работ стало создание поля вычислительной дидактики [digida.mgpu.ru](http://digida.mgpu.ru).

Исходно мы рассматривали вики-площадку как сообщество практики, в котором представители естественно-научных дисциплин могли бы использовать статьи, данные, программы и диаграммы в качестве граничных объектов. На поле образовательной дидактики у нас могут быть представлены: образовательные видеоигры, в которых ученики знакомятся с возможными паттернами поведения программных агентов - исполнителей; обучающие среды (языки программирования), где учащиеся могут создавать свои собственные истории и игры; примеры - рецепты приготовления алгоритмов, управляющих поведением программных агентов. К этим взаимосвязанным классам сущностей следует добавить класс основных понятий поля вычислительной дидактики, класс авторов, которые вводят новые понятия, создают видеоигры и обучающие микромиры; класс роботов исполнителей в окружающем физическом мире вещей; класс сетевых сообществ, в которых может происходить обмен рецептами приготовления алгоритмов; класс знаний и умений, которых требуют образовательные стандарты в сфере вычислительного мышления в разных странах; класс имитационных моделей, которые расширяют содержание книг и учебных курсов.

Объединение и наращивание многообразия сущностей на поле вычислительной дидактики множеством участников предполагает создание разделяемой онтологии,

которая была реализована в среде Semantic MediaWiki через создание категорий - классов объектов. В качестве инструмента коллективного сбора и объединения знаний о поле вычислительной дидактики была использована вики-платформа MediaWiki. В течение нескольких лет мы используем вики как среду для совместной работы со знаниями в образовании. Для изучения области вычислительной дидактики, в которой для обучаемого создаются возможности выступить в роли обучающего для обучаемого программного агента или робота, мы предложили использовать площадку MediaWiki с расширением Semantic MediaWiki [2]. Расширение Semantic MediaWiki позволяет добавлять семантические аннотации к вики-страницам, тем самым превращая вики в семантическую вики и существенно расширяя возможности совместной работы не только над текстами статей, но и над совместной онтологией знаний. Экспериментальная площадка использует технологию Semantic MediaWiki и главная особенность заключается в том, что мы думаем сразу о классах статей объектов, которые относятся к определенной категории и обладают определёнными свойствами.

Мы рассмотрели огромное количество доступных для MediaWiki расширений и установили те, которые существенно расширяют возможности участников по совместной работе со знаниями. В первую очередь это расширения, дополняющие возможности MediaWiki по работе с семантическими свойствами объектов. Кроме того, мы добавили расширение Interwiki, для упрощения работы со статьями с других образовательных вики-площадок. Следующее важное расширение External Data – позволяет использовать на экспериментальной площадке внешние данные из различных источников. Например, различные датасеты, собранные в ходе библиометрических исследований или выращенные в ходе экспериментов с имитационными моделями, открыты для использования и используются на поле совместной деятельности. К работе с данными могут быть отнесены и такие расширения как Scribunto, предназначенного для включения скриптовых языков на страницах MediaWiki. Потенциально это открывает возможность расширить пространство экспериментального поля модулям на языке Lua.

Выбирая средства программирования, которые получили особую поддержку на поле вычислительной дидактики, мы остановили свой выбор на взаимосвязанной группе Scratch, Snap!, Starlogo Nova, NetLogo Web.

Scratch (Snap!) Языки визуального программирования, такие как Scratch, Snap! и многие другие, являются практической альтернативой сложным языкам программирования с текстовым синтаксисом. Они существенно упрощают синтаксис и заменяют слова картинками или блоками. Например, в Scratch блоки, похожие на кирпичики Lego, представляют несколько управляющих конструкций. «Программирование» осуществляется путем перетаскивания и составления этих блоков друг с другом.

StarLogo Nova – среда совместного агентного моделирования в трехмерной среде. Во многом это среда продолжает и развивает традиции языка Scratch. При этом здесь можно ставить серьезные эксперименты и получать воспроизводимые результаты. Центральной метафорой среды является мир (World), в котором можно создавать агенты различных пород. StarLogo Nova предоставляется бесплатно, работает в браузере и дает возможность учащимся делиться своими проектами и играть в игры, созданные другими. Учителя также могут использовать модели, разработанные сообществом StarLogo, для объяснения сложных понятий. Пользователи могут создавать простые или масштабные модели, используя существующую библиотеку агентов или импортируя свои звуки и 3D-модели в формате Collada. Среда активно используется в естественно-научном образовании. В качестве примера следует

привести проект GUTS, в котором взаимосвязаны образовательные стандарты вычислительного мышления и исследовательской деятельности.

NetLogo – это бесплатная мультиагентная программируемая среда моделирования с открытым исходным кодом. По данным сайта NetLogo используют тысячи учеников, преподавателей и исследователей во всем мире. NetLogo написан на Java и Scala. Язык NetLogo достаточно прост, и ученики, и учителя могут создавать в этой среде свои собственные учебные модели. В то же время NetLogo – это достаточно мощный язык для построения исследовательских моделей и проведения исследований в области сложных адаптивных систем. NetLogo удобно использовать для моделирования сложных, развивающихся во времени систем. Создатель модели может давать указания сотням и тысячам независимых «агентов», действующим параллельно. Это открывает возможность для объяснения и понимания связей между поведением отдельных индивидуумов и явлениями, которые происходят на макроуровне.

Для использования внутри экспериментальной площадки по работе со знаниями в области цифрового образования материалов по современным языкам блочного программирования были добавлены два специализированных расширения, позволяющих встраивать визуальные блоки и проекты: ScratchBlocks4 – расширение, которое позволяет встраивать блоки визуального программирования Scratch на страницы вики; Snap! Project Embed – расширение обеспечивает возможность встраивания Snap! проектов на страницы вики.

### Результаты

На поле вычислительной дидактики расширение Semantic MediaWiki позволяет добавлять семантические аннотации к вики-страницам, позволяя совместно работать не только над текстами статей, но и над совместной онтологией знаний. Участники не просто создают и редактируют отдельные статьи, например, про понятие «рекурсия» или «большие данные» или про языки обучения «Scratch» или «Snap!», но и классифицируют статьи, относя их к исходно созданным классам.

Категория языков программирования и сред конструирования была первой и достаточно простой категорией объектов, которые мы начали собирать на поле вычислительной дидактики. Как правило, у языков есть год создания, авторы, языки – предшественники и языки потомки. Даже такой ограниченный перечень параметров позволяет получить историю языков программирования оформленную как ленту времени в ответ на запрос:

```
{{#ask: [[Category:Язык программирования]] [[launch year::+]]
|format=moderntimeline |?launch year |? Ancestors |?Descendants }}
```

Мы можем обращаться к объектам из любого класса и просить их вывести на экран ту или иную информацию. Например, если для категории паттернов мы хотим получить только список с описанием поведения, которое должен имитировать программный агент, то мы обращаемся к системе с запросом:

```
{{#ask: [[Категория:HowTo]] [[Description_of_problem::+]]
?Description_of_problem | format = ol}}
```

Для каждого поведенческого паттерна можно определить видеоигры, где ученики могут посмотреть на примеры поведения исполнителей, которые этот паттерн реализуют. Кроме того, в пространстве языков программирования и образовательных микромиров постепенно накапливаются алгоритмы – рецепты того, как тот или иной паттерн поведения может быть реализован в цифровой среде. Для каждого образовательного микромира мы можем найти или написать программы алгоритмы, реализующие базовые паттерны поведения, с тем чтобы ученики могли их использовать в своих работах в качестве основы. В данном случае использовали в качестве среды

для реализации алгоритмов язык NetLogo, поскольку у языка очень простая нотация и он является современным стандартом при создании как научных, так и учебных симуляций [3,4]. В таблице 1 названия паттернов представлены в первом столбце, а во втором столбце представлен листинг алгоритма, которые позволяет получить необходимый поведенческий паттерн в среде NetLogo.

Таблица 1

Паттерны и алгоритмы	
Описание поведения	Код (NetLogo)
Рождать	<code>crt 10 ask turtles [hatch-sheep 1] ask patches [sprout 10]</code>
Уничтожать	<code>ask turtles with [color = red ] [die]</code>
Сталкиваться	<code>ask turtle 1 [if other turtles-here [die]] ask turtle 2 [if other turtles-here [hatch 2]]</code>
Превращаться в другого	<code>if shape = "ant" [ set shape "ant-has-food"]</code>
Перемещаться случайным образом	<code>to wiggle fd 1 rt random 50 lt random 50 end</code>
Преследовать другого агента	<code>face agent move-to agent set heading</code>
Перевозить другого агента на себе	<code>ask turtle 0 [move-to turtle 1 set heading [heading] of turtle 1]</code>
Толкать (push)	<code>ask turtle 0 [move-to turtle 1 set heading [heading] of turtle 1 fd 2]</code>
Тащить (pull)	<code>ifelse pcolor = black [ set pcolor yellow get-away ]</code>
Распространяться (диффузия)	<code>diffuse chemical (diffusion-rate / 100)</code>
Искать методом восхождения к вершине	<code>uphill patch-variable to move uphill elevation</code>
Подчиняться (клавишам клавиатуры)	<code>to rotate-right if rotate-right-clear? and not game-over? [ ask pieces [ rotate-me-right ] ] end</code>

У объектов из класса паттернов есть так же свойства, связывающие их с образовательными видеоиграми, языками программирования и рецептами - алгоритмами решения той или иной проблемы. Мы можем использовать категорию «Видеоигры» и узнать в какой видеоигре можно посмотреть на реализацию того или иного паттерна или получить информацию из категории «Рецепты» о том, как реализуется то или иное поведение в конкретной среде программирования.

Страницы и свойства Semantic MediaWiki служат строительными блоками-кирпичиками, из которых можно собирать разнообразное содержание. В строительные блоки, которые можно использовать на страницах, добавлены блоки визуального программирования на языках как Scratch и Snap! Например, следующая последовательность текстовых блоков, превращается на странице в визуальные блоки.

<scratchblocks>

когда спрайт нажат

идти (выдать случайное от (20) до (40)) шагов

```

если <касается [край v]>, то
  повернуться к [указатель мышки v]
конец
</scratchblocks>

```

Кроме возможности показывать на странице блоки визуального программирования мы добавили расширения EmbedScratch, Snap! Project Embed, и Widgets:iframe. Использование этих расширений позволяет встраивать в текст любой страницы проекты выполненные и опубликованные на внешних площадках - [scratch.mit.edu](https://scratch.mit.edu), <https://snap.berkeley.edu>, <https://netlogoweb.org/> и <https://www.slnova.org/> Это позволяет учащимся продвигаться от более простых к более сложным примерам реализации решения сходных проблем в различных средах многоагентного программирования.

Связь понятий, представленных на поле цифровой дидактики, может раскрываться за счёт использования программных средств, которые используют концептуальные особенности понятий, и имитационных моделей, которые позволяют раскрыть содержание понятия. Например, для понятия «Зона ближайшего развития» в сети его отношений значение будут иметь и исходящие, и приходящие к этому понятию ссылки. Кроме того, смысл этого понятия раскрывается на площадке через многоагентную модель, в которой имитируются различные стратегии обучения [5]. Фрагмент листинга «обучения по Выготскому» представлен ниже:

```

to v-adjust
  let fellow nobody
  while [ fellow = nobody or fellow = self ] [
    set fellow turtle (who + (- (#-vygotskiiian-neighbors / 2)) + random (1 + #-vygotskiiian-neighbors))
  ]
  ifelse (best-score > [ best-score ] of fellow) and (best-score - ZPD <= [ best-score ] of fellow) [
    set best-score [ best-score ] of fellow
    set best-max-moves [ best-max-moves ] of fellow ]
  [ set best-score score
    set best-max-moves max-moves
  ]
end

```

Еще одно направление расширения выразительных возможностей вики площадки связано с совместным использованием блоков визуального программирования. Расширение поддерживает прежде всего представление блоков языка Scratch, однако может быть использовано и для языков Snap! и GP – и в связке с возможностью встраивать в вики-статьи визуальный код программ и выполняемые проекты с сайта Snap! это даёт участникам возможность обмениваться материалами и выполняемыми публикациями.

### Выводы

Среда Semantic MediaWiki открывает возможности не только для размещения разнообразного материала, но и для изучения его дальнейшего использования. Перечень моделей с описанием можно получить по запросу `{{#ask: [[Category:Model]] | ?Description }}`. Число моделей получается по запросу `{{#ask: [[Category:Model]] | format=count }}`

В контексте обсуждения воздействия имитационных моделей на поле вычислительной дидактики значение имеет повторное использование моделей внутри

других статей. Эти ссылки и включения из других страниц мы можем получить для каждой модели через запрос "Службная:Ссылки\_сюда"

Например, для модели [http://digida.mgpru.ru/index.php/Flocking\\_\(model\)](http://digida.mgpru.ru/index.php/Flocking_(model)) обратные ссылки можно получить по запросу

[http://digida.mgpru.ru/Special:WhatLinksHere/Flocking\\_\(model\)](http://digida.mgpru.ru/Special:WhatLinksHere/Flocking_(model))

При этом нас интересуют прежде всего ссылки на имитационные модели из страниц, которые принадлежат к категории "События", в которой собирается информация о мастер-классах, лекциях и семинарах, которые проходят на основе материалов, представленных в среде Semantic MediaWiki.

Запрос к категории События с просьбой выделить события, в рамках которых использовались имитационные модели, позволил получить перечень из, которого следует, что более половины всех событий, которые происходили на площадке [digida.mgpru.ru](http://digida.mgpru.ru), были поддержаны использованием имитационных моделей. Среди таких событий:

- Development of computational thinking based on collective interaction in MediaWiki and multi-agent approach
- Models & Communities,
- аналитика совместной деятельности и поведения отдельных пользователей,
- генеративный искусственный интеллект – новая Черепашка, помогающая думать,
- генерация учебных задач при помощи генеративных моделей,
- использование генеративного искусственного интеллекта для формирования вычислительного мышления школьников,
- использование сред агентного моделирования для изучения городской среды
- коллективное представление знаний внутри систем медицинского образования,
- объединение языков многоагентного моделирования в учебном процессе на базе MediaWiki,
- путь черепахи: эволюция LOGO-подобных языков.

### Литература

1. Minsky M., Papert S.A. Perceptrons: An Introduction to Computational Geometry. MIT Press, 2017. 317 p.
2. Патаракин Е.Д. Игровое поле вычислительной дидактики // Современная “цифровая” дидактика. М.: ГринПринт, 2022. P. 35–70.
3. Wilensky U., Rand W. An Introduction to Agent-Based Modeling: Modeling Natural, Social, and Engineered Complex Systems with NetLogo. MIT Press, 2015.
4. Railsback S.F., Grimm V. Agent-Based and Individual-Based Modeling: A Practical Introduction, Second Edition. Princeton University Press, 2019. 359 p.
5. Abrahamson D., Wilensky U. Piaget? Vygotsky? I’m Game!—Agent-Based Modeling for Psychology Research. 2005.