

DOI: 10.32517/2221-1993-2024-23-4-23-32

И. А. Быкова*Ярославский государственный педагогический университет им. К. Д. Ушинского, г. Ярославль, Россия;
Средняя школа № 26, г. Ярославль, Россия***М. Л. Морозов***Ярославский государственный педагогический университет им. К. Д. Ушинского, г. Ярославль, Россия*

РАЗРАБОТКА И АНАЛИЗ МОДЕЛИ ДОРОЖНОГО ДВИЖЕНИЯ В СРЕДЕ ANYLOGIC*

Аннотация

В федеральной рабочей программе по информатике для XI класса (углубленный уровень) указано на необходимость рассмотрения вопросов построения имитационных моделей, предусмотрено выполнение практических работ на моделирование движения и обработку результатов эксперимента. Интересным и понятным для школьников является разработка и анализ имитационных моделей, связанных с организацией дорожного движения. Реализовать указанные модели можно в среде AnyLogic, для чего целесообразно предложить школьникам построить модель знакомого им перекрестка и проанализировать: при какой интенсивности движения транспорта нет необходимости устанавливать светофор, когда светофор необходим, какие временные фазы работы светофора обеспечивают наибольшую пропускную способность перекрестка. В статье описаны алгоритмы создания моделей нерегулируемого и регулируемого перекрестков; представлены задания на анализ пропускной способности перекрестка при различных значениях количества автомобилей на каждом направлении движения в час; даны задания на оптимизацию фаз работы светофора для увеличения пропускной способности перекрестка; приведены примеры тем проектов для организации самостоятельной работы учащихся.

Ключевые слова: имитационное моделирование, обработка результатов эксперимента, преподавание информатики, AnyLogic.

1. Введение

Тема «Моделирование» занимает одно из центральных мест в школьном курсе информатики. Изучение данной темы направлено на развитие системного мышления учащихся, позволяет осуществлять более эффек-

тивное освоение школьниками основ работы с информационными технологиями. В УМК И. А. Калинина, Н. Н. Самылкиной [2–4] сделан акцент на освоении общенаучного системного подхода в моделировании, реализован переход от математического моделирования к моделированию систем и процессов с использованием

* Материалы к статье можно скачать на сайте ИНФО: http://infojournal.ru/journals/school/school_04-2024/

Контактная информация

Быкова Ирина Альбертовна,

старший преподаватель кафедры теории и методики обучения информатике, Ярославский государственный педагогический университет им. К. Д. Ушинского, г. Ярославль, Россия; *адрес:* 150000, Россия, г. Ярославль, ул. Республиканская, д. 108/1; учитель информатики, средняя школа № 26, г. Ярославль, Россия; *адрес:* 150045, Россия, г. Ярославль, ул. Блюхера, д. 74; *e-mail:* i.bukova@yandex.ru

Морозов Максим Леонидович,

магистрант кафедры теории и методики обучения информатике, Ярославский государственный педагогический университет им. К. Д. Ушинского, г. Ярославль, Россия; *адрес:* 150000, Россия, г. Ярославль, ул. Республиканская, д. 108/1; *e-mail:* maximmorozo@yandex.ru

I. A. Bikova

Yaroslavl State Pedagogical University named after K. D. Ushinsky, Yaroslavl, Russia;
School 26, Yaroslavl, Russia

M. L. Morozov

Yaroslavl State Pedagogical University named after K. D. Ushinsky, Yaroslavl, Russia

DEVELOPMENT AND ANALYSIS OF A TRAFFIC MODEL IN THE ANYLOGIC ENVIRONMENT

Abstract

The Federal Work Program on Informatics for the 11th grade (advanced level) indicates the need to consider the issues of constructing simulation models, provides the practical works on motion modeling and processing of experimental results. Development and analysis of simulation models related to road traffic management is interesting and understandable for schoolchildren. These models can be implemented in the AnyLogic environment, for which it is advisable to offer schoolchildren to create a model of an intersection they are familiar with and analyze at what traffic intensity there is no need to install a traffic light, when a traffic light is necessary, what time phases of the traffic light provide the highest capacity of the intersection. The article describes algorithms for creating models of unregulated and regulated intersections; presents tasks for analyzing the capacity of the intersection for different values of the number of cars in each direction of traffic per hour; describes tasks for optimizing the phases of the traffic light to increase the capacity of the intersection; provides examples of project topics for organizing independent work of students.

Keywords: simulation modeling, processing experimental results, teaching informatics, AnyLogic.

системы многопроходного имитационного моделирования [6–8].

Имитационная модель — это компьютерная программа, которая описывает структуру реальной системы и воспроизводит ее поведение во времени. **Имитационное моделирование** — процесс разработки компьютерных моделей и постановки экспериментов с их использованием. Целью моделирования, в конечном счете, является принятие обоснованных, целесообразных управленческих решений [1, 5].

Для организации практических работ авторы УМК [2–4] предлагают использовать **бесплатную образовательную версию среды AnyLogic — Personal Learning Edition**, которая дает возможность создавать, демонстрировать и исследовать широкий спектр моделей из самых разных областей деятельности. Использование этой среды позволяет продемонстрировать важность методов моделирования, их возможности при решении практических задач [10, 11]. С помощью AnyLogic можно разрабатывать модели в различных областях: логистика, рынок и конкуренция, бизнес-процессы и сфера обслуживания, социальные и экологические системы, дорожное движение и т. д.

AnyLogic позволяет реализовать две фазы моделирования:

- разработка имитационной модели;
- последующий прогон модели для анализа.

Интересным и понятным для школьников является разработка и анализ имитационных моделей, связанных с организацией дорожного движения, для чего в среде AnyLogic имеется специальная библиотека.

В федеральной рабочей программе по информатике для XI класса указано на необходимость рассмотрения вопросов построения имитационных моделей на основе вероятностных моделей, предусмотрено выполнение практических работ на моделирование движения и обработку результатов эксперимента [9]. При реализации указанных практических работ можно предложить школьникам построить модель знакомого им перекрестка и проанализировать:

- при какой интенсивности движения транспорта нет необходимости устанавливать светофор;
- когда светофор необходим;
- какие временные фазы работы светофора обеспечивают наибольшую пропускную способность перекрестка.

2. Этапы разработки и анализа модели дорожного движения

Главная цель построения модели транспортной сети — оптимизация пропускной способности данного участка дороги при заданных условиях: количество автомобилей на каждом направлении движения в час. Учитывая эти условия, необходимо отрегулировать время смены сигналов светофора.

Организация познавательной деятельности учащихся может быть следующей:

1. Обсуждение цели построения модели. Знакомство с интерфейсом программы AnyLogic.
2. Построение модели нерегулируемого перекрестка.
3. Работа с моделью нерегулируемого перекрестка (анализ пропускной способности перекрестка при

различных значениях количества автомобилей на каждом направлении движения в час).

4. Построение модели регулируемого перекрестка.
5. Работа с моделью регулируемого перекрестка (оптимизация фаз работы светофора для увеличения пропускной способности перекрестка).
6. Обсуждение результатов работы и возможных тем проектов для самостоятельной работы.

Рассмотрим подробнее некоторые из указанных этапов.

3. Построение модели нерегулируемого перекрестка

Для создания модели нерегулируемого перекрестка можно использовать описанный ниже алгоритм, который представлен в виде пошаговой инструкции для учащихся.

Заметим, что иллюстрации шагов алгоритма относятся к реальному перекрестку улиц города Ярославля — Республиканской (двусторонняя дорога с одной полосой движения в каждом направлении) и Большой Октябрьской (двусторонняя дорога с двумя полосами движения в каждом направлении).

3.1. Шаг 1. Создание новой модели

Выберите команду *Создать, Новая модель*. В появившемся окне задайте имя модели и нажмите кнопку *Готово*.

3.2. Шаг 2. Добавление карты

Из библиотеки *Презентация* перетащите на рабочее поле элемент *Изображение*. Выберите файл с изображением перекрестка на вашем компьютере (*учителю необходимо заранее подготовить файл с изображением нужного перекрестка*). Расположите карту так, чтобы вам было удобно с ней работать. В свойствах изображения поставьте галочку около элемента *Блокировать*.

3.3. Шаг 3. Построение главной дороги

В библиотеке *Дорожное движение* выберите элемент *Дорога*. Двойной щелчок левой кнопкой мыши по данному элементу включает режим рисования — нарисуйте дорогу там, где она должна проходить. Операция рисования дороги завершается также двойным щелчком левой кнопкой мыши.

В свойствах дороги укажите количество полос основного и встречного движения (рис. 1).

3.4. Шаг 4. Построение второстепенной дороги и перекрестка

Аналогично добавьте первую часть второстепенной дороги, местом окончания укажите уже построенную дорогу. На этом месте автоматически должен появиться перекресток (рис. 2).

Добавьте вторую часть второстепенной дороги, местом начала укажите уже построенную дорогу.

В свойствах дорог укажите количество полос движения.

На рисунке 3 изображена дорожная сеть, состоящая из перекрестка и четырех дорог с двусторонним движением: две дороги — с двумя полосами движения, две дороги — с одной полосой движения в каждую сторону.

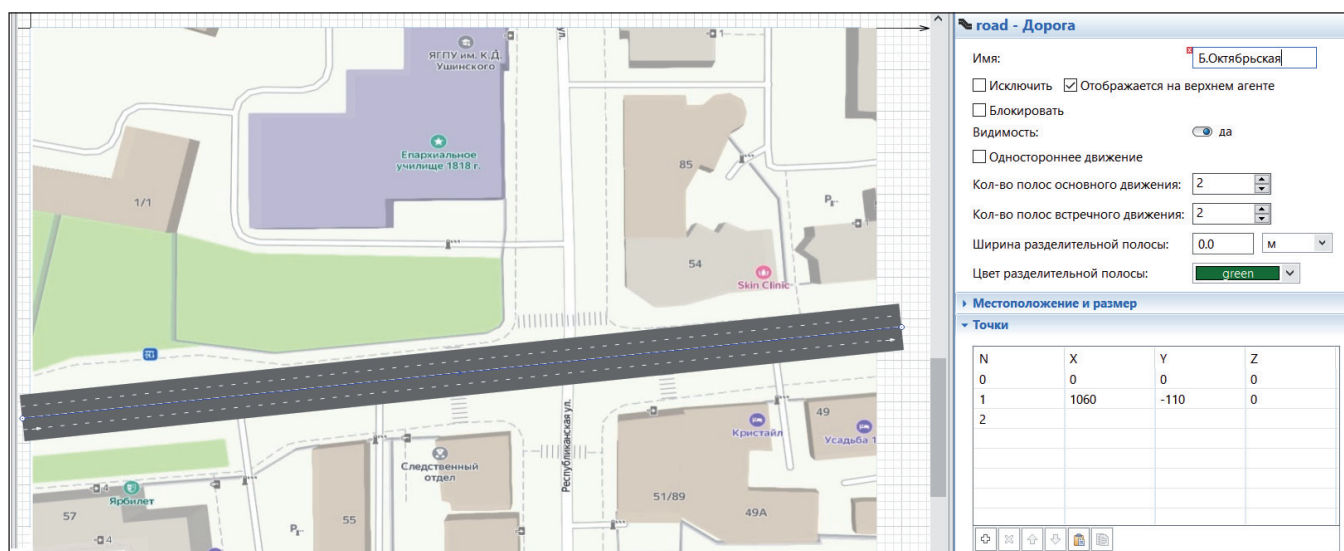


Рис. 1. Построение главной дороги на улице Большой Октябрьской



Рис. 2. Построение второстепенной дороги на улице Республиканской

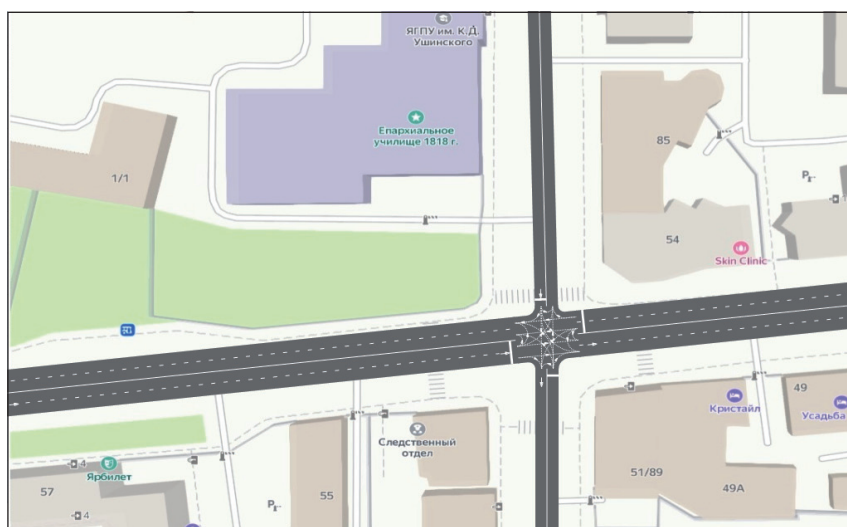


Рис. 3. Результат построения дорожной сети

3.5. Шаг 5. Создание автомобиля

Из библиотеки дорожного движения перетащите на рабочую зону элемент *Тип автомобиля*.

В появившемся диалоговом окне укажите название автомобиля — в примере на рисунке 4 «Авто1».

Затем выберите анимацию — пункт *3D*.

После этого создайте параметр с именем *Время-Появления* — он понадобится для анализа пропускной способности построенной сети дорог.

В результате на рабочем поле появится еще одна вкладка с именем элемента.

3.6. Шаг 6. Создание логики движения автомобиля

Из библиотеки дорожного движения перетащите на рабочую зону элемент *CarSource* (он будет генерировать автомобили).

Укажите в его свойствах (рис. 5):

- место появления агентов;
- тип появления агентов (*Прибывают согласно*);
- интенсивность прибытия.

Добавьте элемент *CarMoveTo* (цель движения), привяжите его к предыдущему элементу. В свойстве *Дорога*

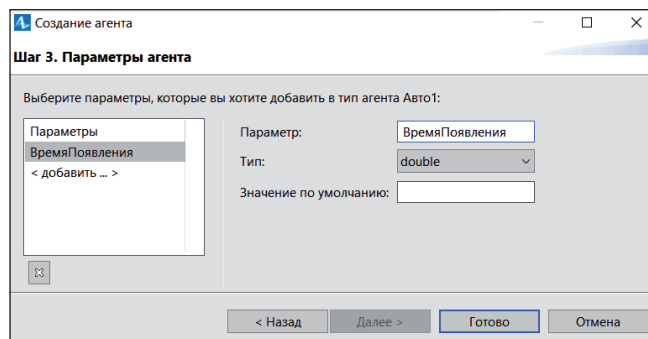
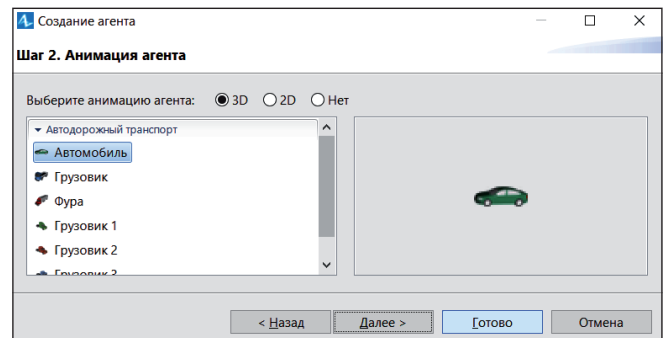
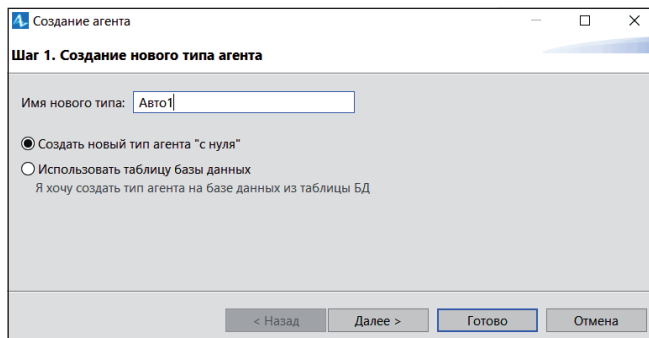


Рис. 4. Настройка элемента *Автомобиль*

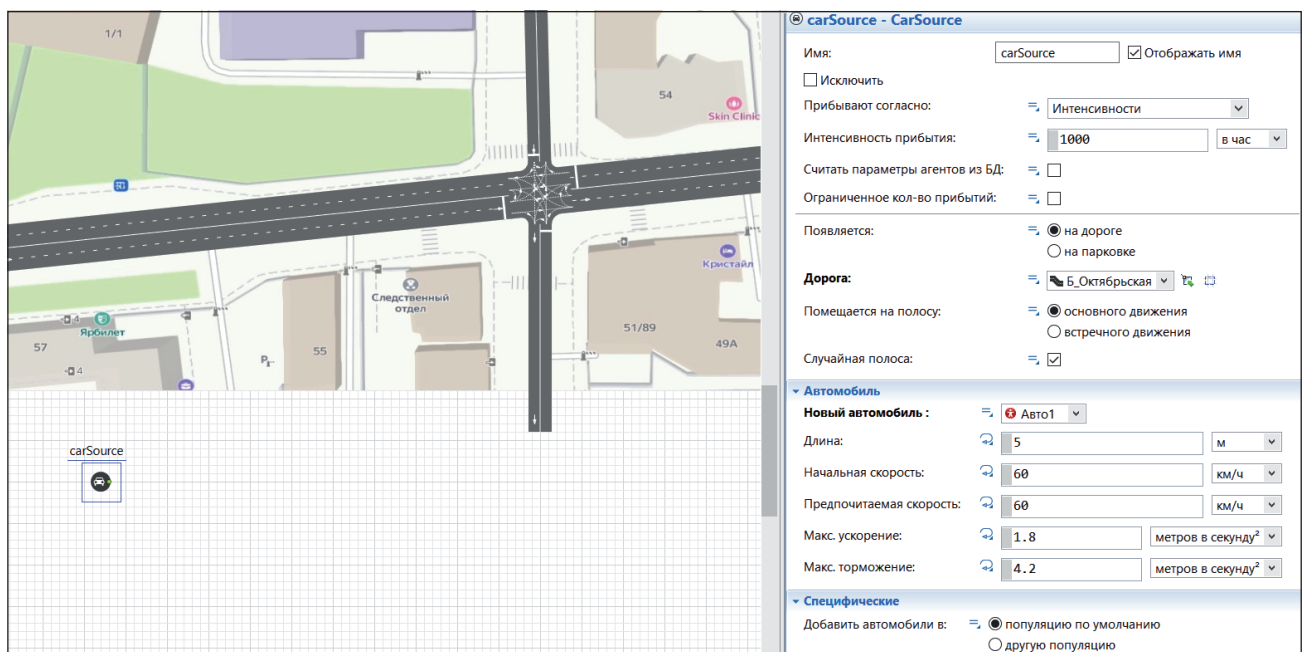


Рис. 5. Настройка начала движения

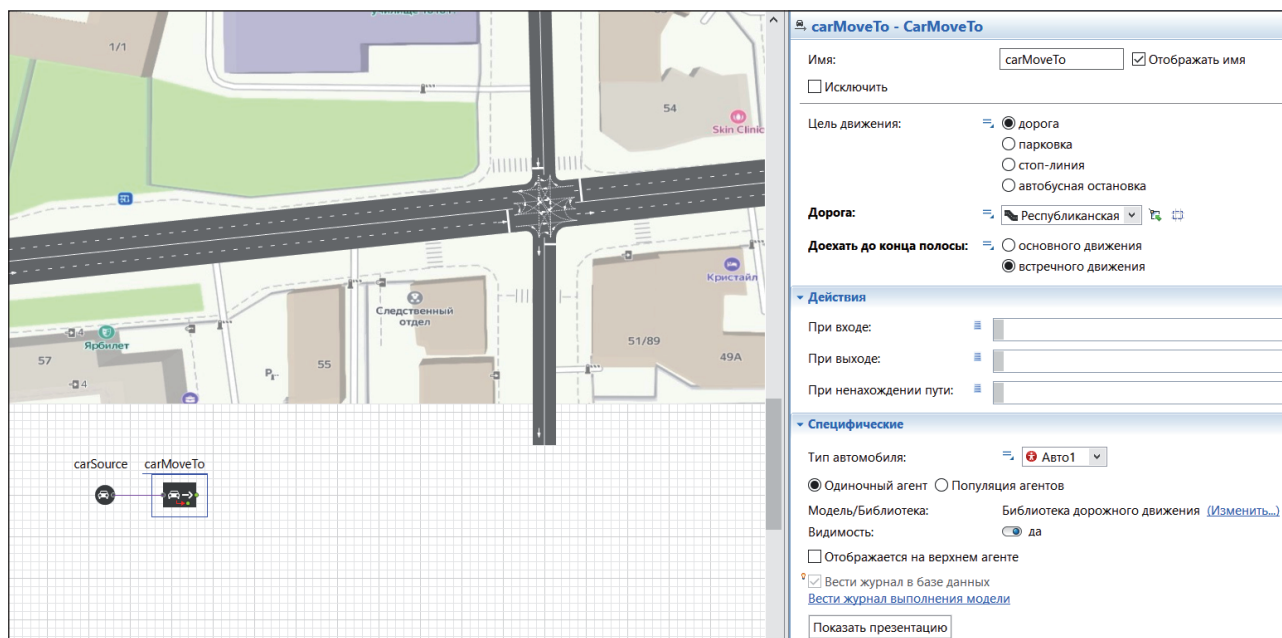


Рис. 6. Настройка направления проезда перекрестка

данного элемента укажите, куда будут двигаться агенты после прохождения перекрестка.

Обратите внимание на полосу движения (основное или встречное) (рис. 6).

Добавьте элемент *CarDispose* (удаление автомобилей), привяжите его к предыдущему элементу.

3.7. Шаг 7. Запуск модели

Запустите модель на выполнение — для этого во вкладке верхнего меню *Модель* выберите пункт *Запустить* или нажмите клавишу F5. В новом окне откроется модель построенного перекрестка, на которой будут двигаться автомобили только по одной траектории (рис. 7).

3.8. Дополнительно

Для моделирования движения транспорта в различных направлениях можно выполнить шаг 6 несколько раз для каждого направления движения, а можно сделать выбор случайным с заданным значением вероятности с помощью элемента *SelectOutput5*.

Добавьте элемент *SelectOutput5* из библиотеки процессов между элементами *CarSource* и *CarMoveTo*, создайте связи между всеми возможными способами проезда перекрестка (рис. 8), укажите вероятности выбора отдельных дорог. Создание связи начинается с двойного щелчка мышью по порту начала связи, изменение направления связи и завершение связи осуществляются с помощью одинарного щелчка левой кнопкой мыши.

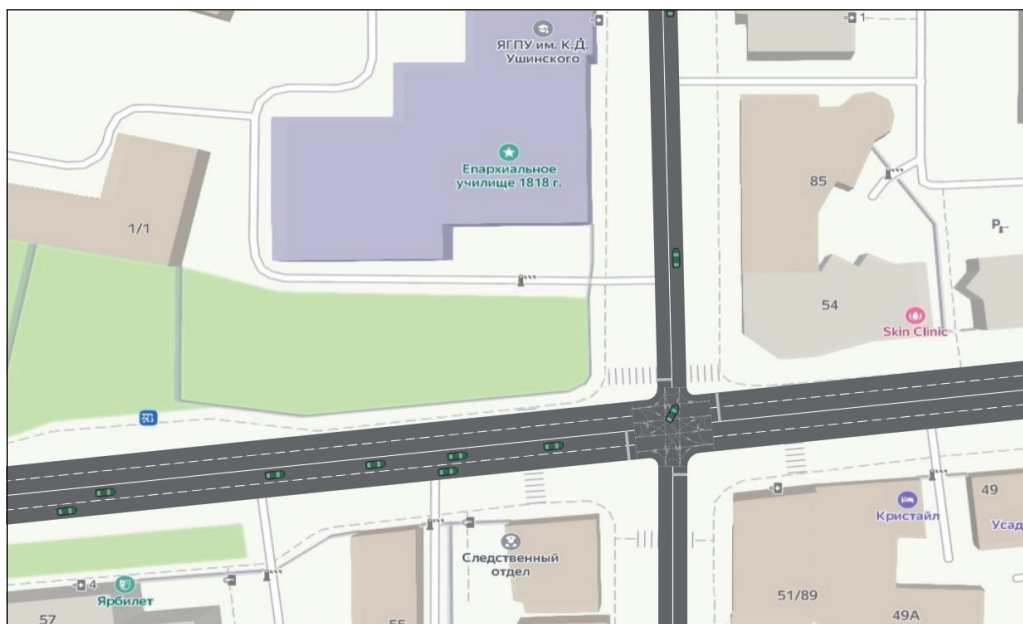


Рис. 7. Выполнение модели

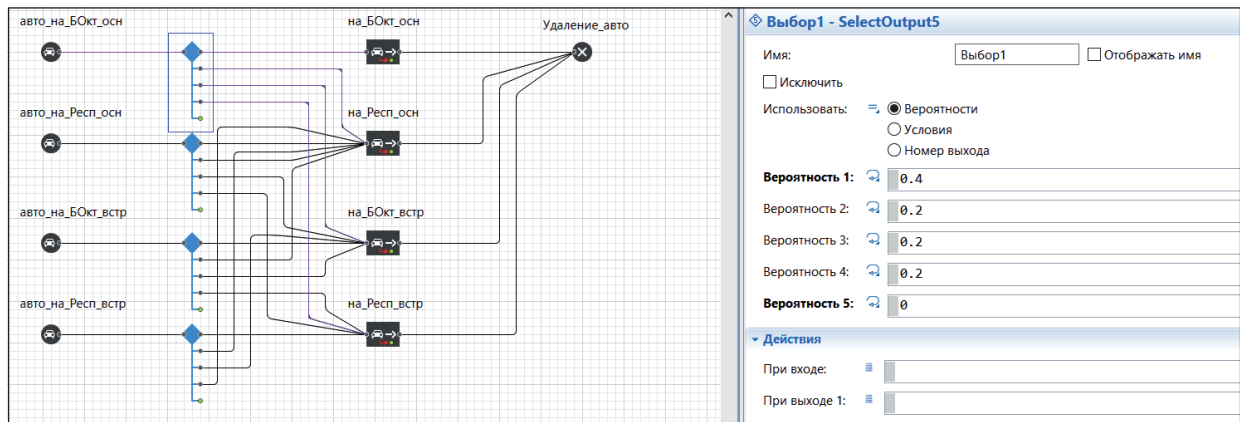


Рис. 8. Настройка возможных направлений движения

4. Анализ модели нерегулируемого перекрестка

Для анализа модели нерегулируемого перекрестка можно использовать следующие **задания**:

Задание 1.

Укажите, при какой интенсивности прибытия объектов на дорогах начнут образовываться пробки (рис. 9):

- 1000 объектов в секунду;
- 2000 объектов в секунду;
- 500 объектов в секунду.

Задание 2.

Определите максимальную интенсивность, при которой образование пробки не происходит.

5. Построение модели регулируемого перекрестка

Для создания светофора добавьте элемент *TrafficLight* из библиотеки дорожного движения к месту перекрестка.

В свойствах объекта укажите перекресток регулирования, время и фазы проезда (рис. 10).

6. Анализ модели регулируемого перекрестка с помощью диаграммы

Для получения первичной статистической информации необходимо создать популяцию агентов и добавить в модель диаграмму, в которой будет отображена информация о том, какой процент машин проходит перекресток за выделенные интервалы времени. Во время работы модели на диаграмме будут отображаться указанные в настройках характеристики.

6.1. Шаг 1. Создание популяции агентов

В библиотеке *Агент* выберите элемент *Агент* и перетащите его на рабочую область проекта.

В открывшемся диалоговом окне выберите пункт *Популяция агентов*, измените имя популяции на «Автомобили».

Перейдите до вкладки *Параметр*, выберите ранее созданный параметр *ВремяПоявления*.

Выберите *Пустая популяция* и нажмите кнопку *Готово*.

Во всех элементах *CarSource* измените свойство агентов — в разделе *Специфические* выберите пункт *Добавить автомобили в, другую популяцию*, укажите имя популяции (рис. 11).



Рис. 9. Пример образования пробки на дороге

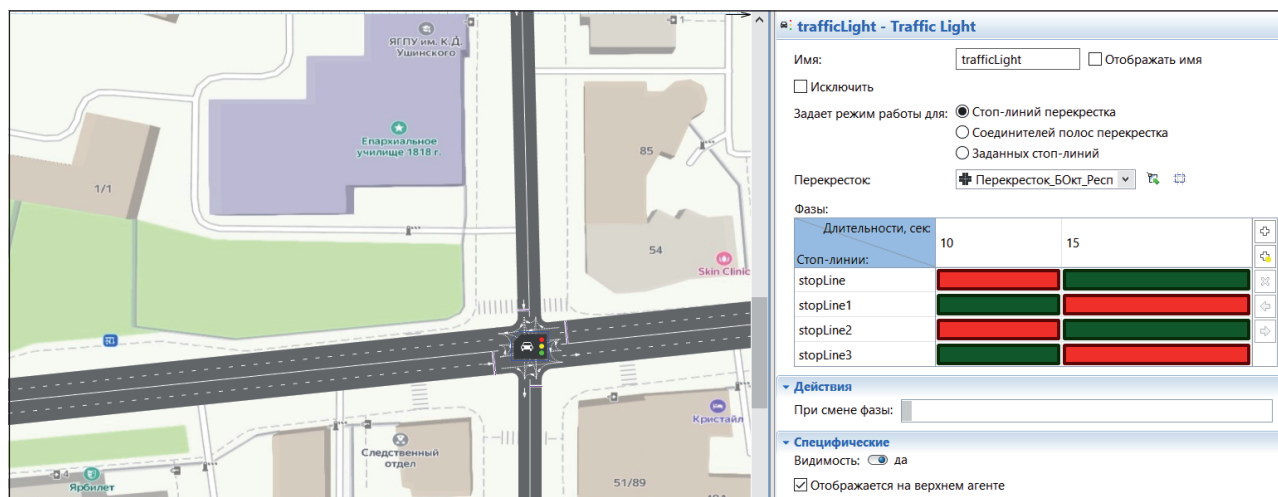
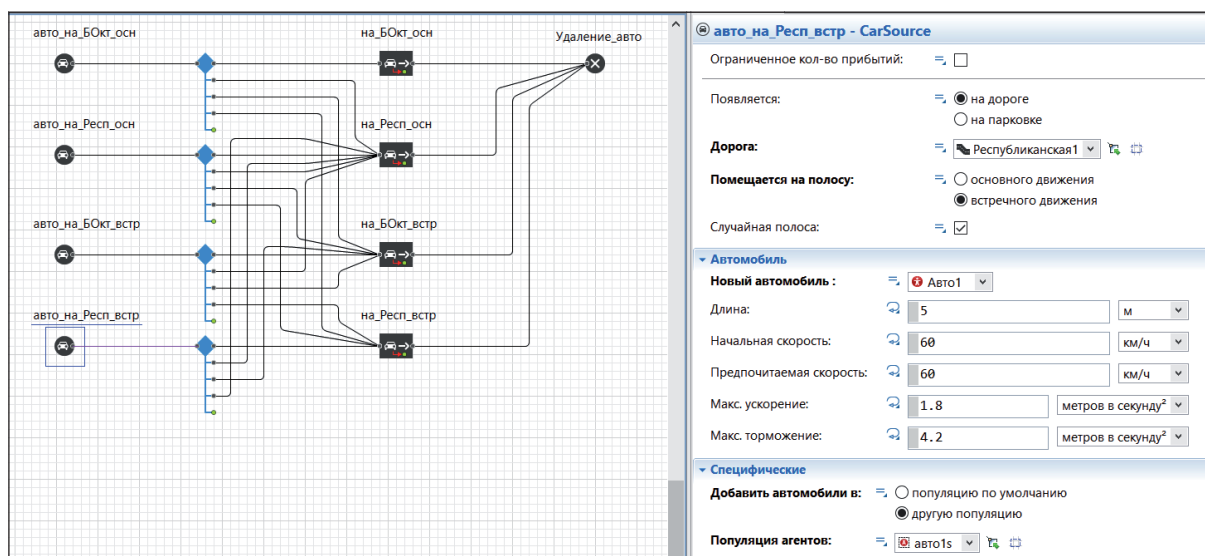
Рис. 10. Настройка элемента *Перекресток*

Рис. 11. Добавление агентов в популяцию

6.2. Шаг 2. Создание гистограммы

Из библиотеки *Статистика* перетащите на рабочую область элемент *Данные гистограммы*. Измените имя элемента на «ВремяПроезда».

Перейдите к вкладке агента «Авто1» (рис. 12). В разделе *Предв. просмотр параметров* выполните двойной щелчок левой кнопкой мыши, в появившемся окне в поле *Значение по умолчанию* укажите команду: `time()`. В результате значение параметра *ВремяПоявления* будет равно текущему времени.

В разделе *Действия агента* в поле *При уничтожении* укажите код: `main.ВремяПроезда.add(time()-ВремяПоявления);` (рис. 13). Данный код фиксирует время между появлением агента и временем удаления его из системы, тем самым передает в качестве данных диаграммы список времен проезда перекрестка отдельными агентами.

Из библиотеки *Статистика* перетащите на рабочую зону элемент *Гистограмма*, в свойстве *Данные* добавьте имя элемента — «ВремяПроезда» (рис. 14).

Запустите модель, посмотрите, как меняется диаграмма с течением времени (рис. 15).

7. Анализ модели регулируемого перекрестка с использованием возможности оптимизации модели AnyLogic

Найти значения параметров, при которых достигается наилучший результат работы модели, можно, используя возможность оптимизации модели AnyLogic. Оптимизация модели AnyLogic заключается в последовательном выполнении нескольких прогонов модели с различными значениями параметров и нахождении оптимальных для данной задачи значений параметров.

7.1. Шаг 1. Настройка параметров фаз работы светофора

Из библиотеки *Агент* перетащите на рабочую зону два элемента *Параметр*, дайте им имена «p1» и «p2», значения по умолчанию — 30 и 15 соответственно. Укажите имена параметров вместо времени фаз в меню настройки светофора (рис. 16).

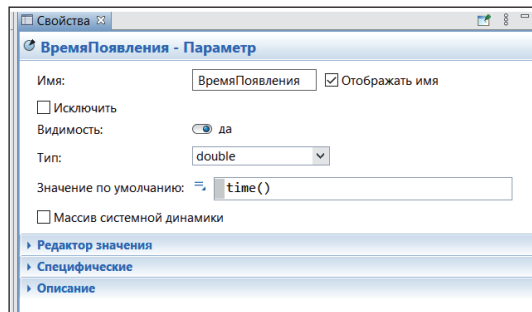


Рис. 12. Настройка времени появления агента

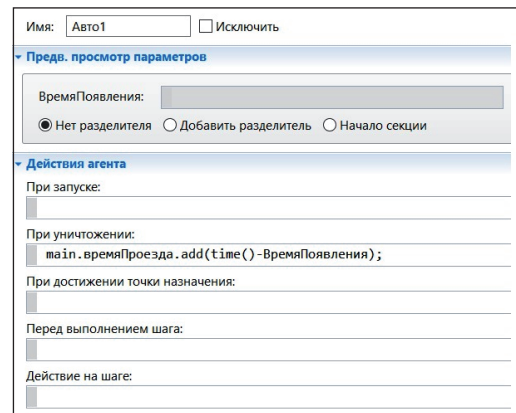


Рис. 13. Настройка параметров действий агента

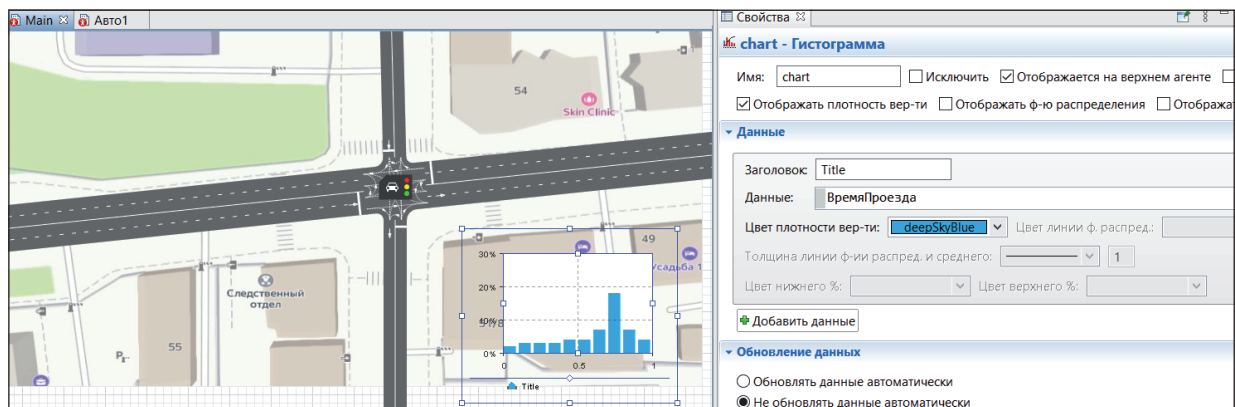


Рис. 14. Настройка элемента Гистограмма

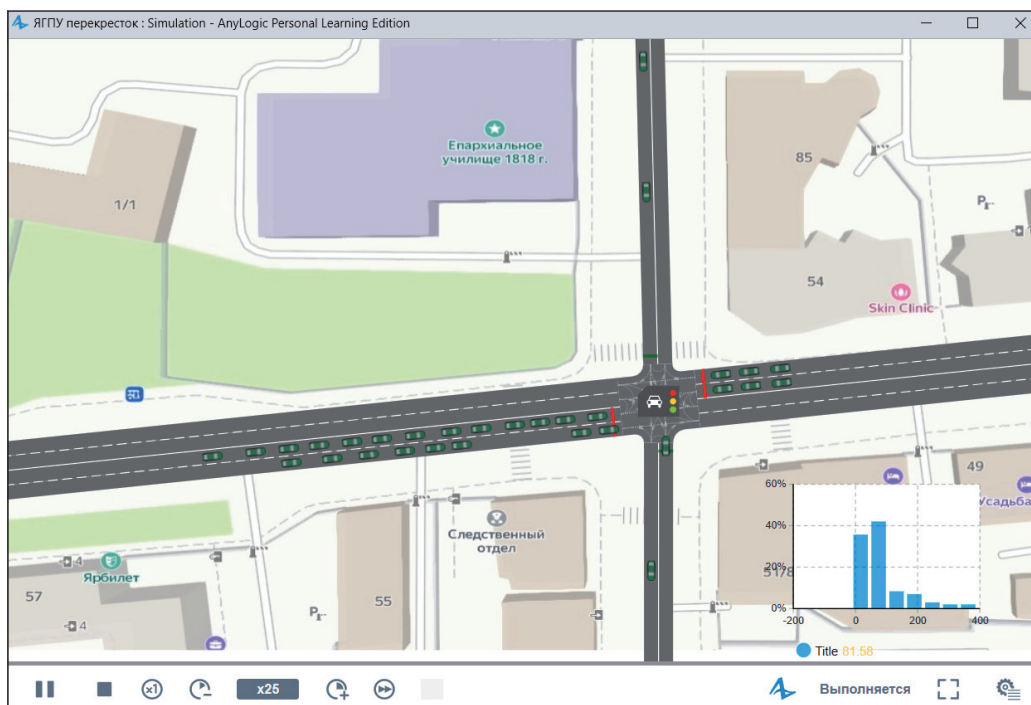


Рис. 15. Окно работающей модели

7.2. Шаг 2. Организация эксперимента

Во вкладке *Проекты* вызовите контекстное меню для обрабатываемой модели и выберите в нем команду

Создать, Эксперимент, в появившемся диалоговом окне выберите пункт *Оптимизация*. В рабочей зоне появится новая вкладка с именем эксперимента (рис. 17).

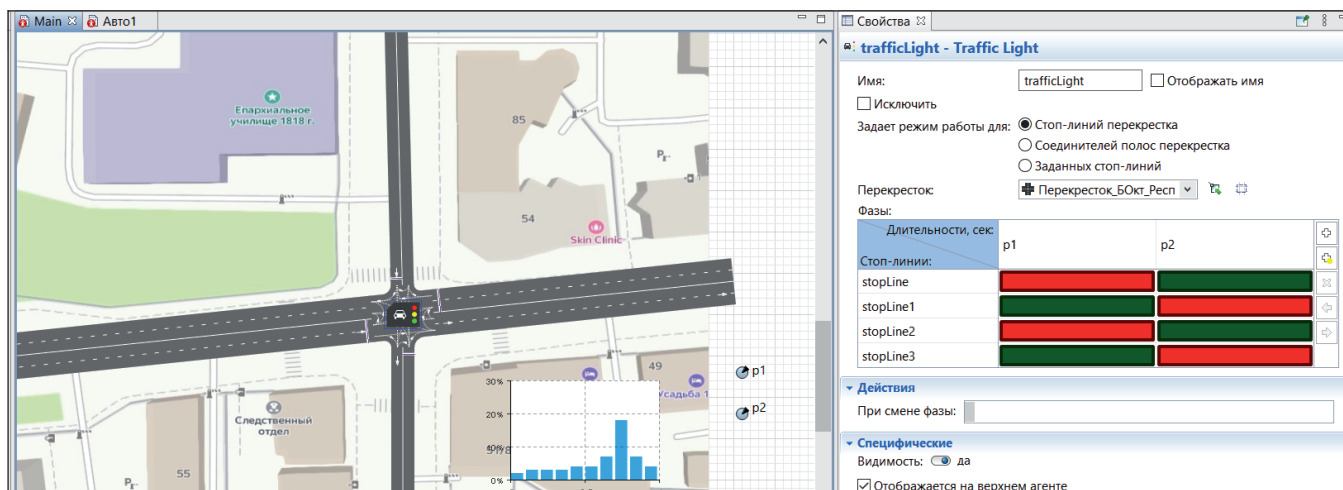


Рис. 16. Настройка параметров фаз работы светофора

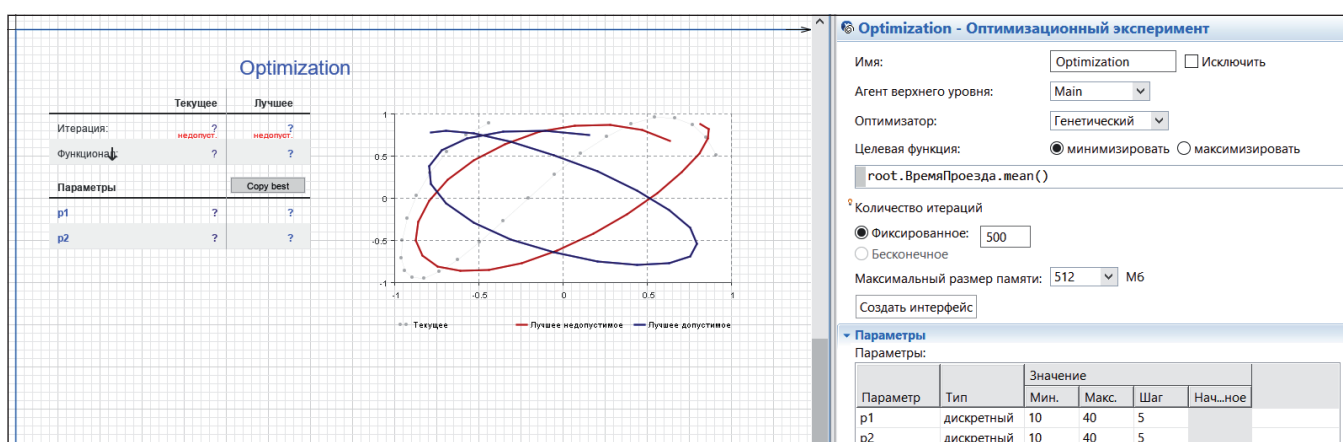


Рис. 17. Настройка параметров эксперимента

В открывшемся меню укажите в качестве целевой функции следующее значение: `root.ВремяПроезда.mean()`.

Измените тип параметра на *дискретный* и задайте границы изменения параметра: минимальное значение для каждого — 10, максимальное — 40, шаг — 5. Нажмите кнопку *Создать интерфейс*.

7.3. Шаг 3. Запуск эксперимента

Запустите модель на выполнение: во вкладке верхнего меню *Модель* выберите пункт *Запустить* и в появившейся вкладке выберите пункт с названием эксперимента. Откроется окно эксперимента (рис. 18).

Сделайте вывод об оптимальном времени различных фаз светофора при заданной интенсивности потока автомобилей.

Измените интенсивность, проведите повторный эксперимент.

Добавьте «желтые» фазы работы светофора, для них задайте два параметра. Повторите эксперимент.

8. Рекомендации для организации самостоятельной работы учащихся

Школьникам, проявившим интерес к изучению темы, можно предложить самостоятельно реализовать один из представленных ниже проектов.

Проект 1. Самый «удобный» перекресток.

Цель проекта: определение самого «удобного» перекрестка с точки зрения наибольшей проходимости.

Учащимся необходимо проанализировать несколько видов перекрестков: крестообразный, Т-образный, У-образный. Частота агентов для всех направлений остается стандартной, т. е. не задается оператором.

Выводы по данному проекту делаются за счет встроенных элементов статистики, которые позволяют определять наибольшую проходимость за заданное время.

Проект 2. Мне не нравится ждать.

Цель проекта: определение самой свободной дороги с точки зрения заторов и пробок на дороге.

Учащимся необходимо проанализировать несколько видов перекрестков: крестообразный, Т-образный, У-образный. Для данного проекта количество агентов задается оператором (имеется возможность использования данных реального мира).

Выводы делаются за счет встроенных элементов определения загруженности дорог.

Проект 3. Сеть общественного транспорта.

Цель проекта: создание модели общественного транспорта.

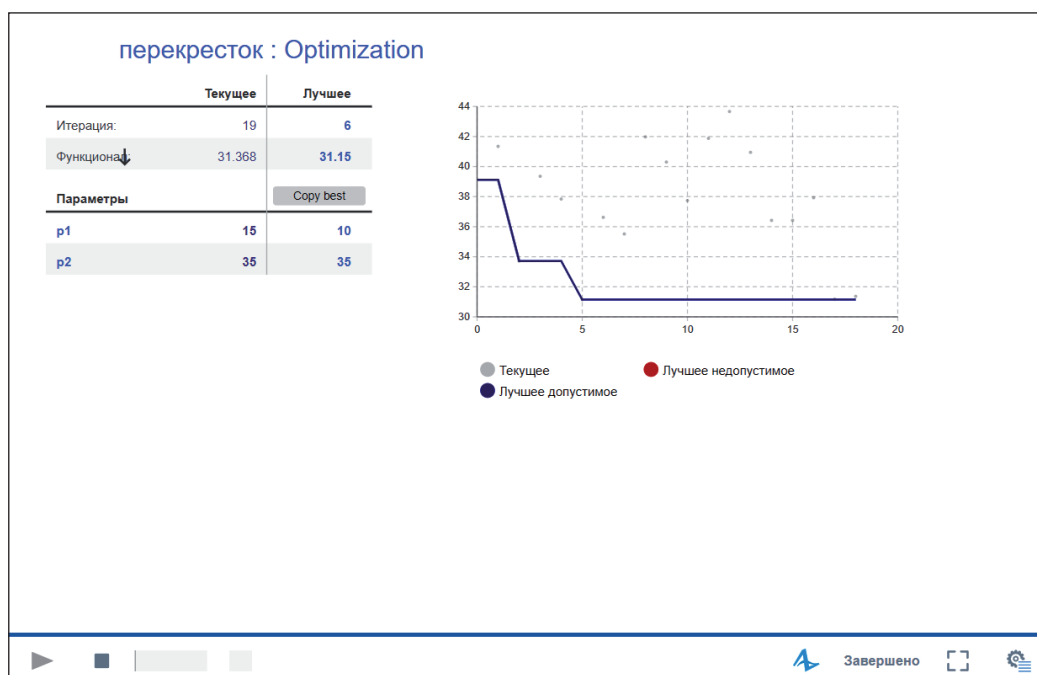


Рис. 18. Окно эксперимента

Учащимся необходимо дополнить описанную выше модель новыми агентами — автобусами и остановками общественного транспорта.

Проект 4. Точь-в-точь.

Цель проекта: как можно более точно перенести окружающий мир на модель в рамках одного перекрестка.

Учащимся необходимо использовать дополнительные возможности работы с 3D-объектами.

9. Выводы

Содержание курса информатики для старшеклассников, по мнению Н. Н. Самылкиной, должно раскрывать реальное состояние отрасли информационных технологий и направлений ее развития; быть современным с точки зрения использования инструментов ИТ и подходов к их изучению с позиции планируемых результатов образовательной деятельности [7, 8]. Решение описанных в статье практических задач при изучении темы «Моделирование» позволяет формировать общую культуру моделирования и принятия решений.

Описанные в статье модели могут быть использованы при изучении темы «Имитационное моделирование» в старших классах средней школы. Дополнительно можно предложить школьникам расширить функционал построенных моделей за счет использования пешеходной библиотеки и других возможностей AnyLogic.

Список источников

1. Калинин И. А., Самылкина Н. Н. Имитационное моделирование // Информатика. 2013. № 5. С. 4–17. EDN: VNWJG.
2. Калинин И. А., Самылкина Н. Н. Информатика. 10 класс. Углубленный уровень: учебник. М.: БИНОМ. Лаборатория знаний, 2019. 256 с. EDN: XHXXSW.

3. Калинин И. А., Самылкина Н. Н. Информатика. 11 класс. Углубленный уровень: учебник. М.: БИНОМ. Лаборатория знаний, 2018. 211 с. EDN: YJWQEK.

4. Калинин И. А., Самылкина Н. Н., Бочаров П. В. Информатика. Углубленный уровень: задачник-практикум для 10–11 классов. М.: БИНОМ. Лаборатория знаний, 2015. 248 с. EDN: GJCUXL.

5. Киселева М. В. Имитационное моделирование систем в среде AnyLogic: учебно-методическое пособие. Екатеринбург: УГТУ—УПИ, 2009. 88 с.

6. Самылкина Н. Н. Дидактический потенциал среды имитационного моделирования AnyLogic для решения разных типов управленческих задач // Информатизация непрерывного образования — 2018. Материалы Международной научной конференции (Москва, 14–17 октября 2018 года). В 2 т. Т. 1. М.: РУДН, 2018. С. 381–385. EDN: YIKBMR.

7. Самылкина Н. Н. Организация углубленного обучения информатике на основе интегративного подхода: монография. М.: МПГУ, 2020. 346 с. EDN: YZMPYM.

8. Самылкина Н. Н., Калинин И. А. Новый взгляд на информатику: имитационное моделирование, искусственный интеллект и блокчейн в углубленном курсе информатики // Актуальные проблемы методики обучения информатике и математике в современной школе. Материалы международной научно-практической интернет-конференции (Москва, 22–26 апреля 2019 года). М.: МПГУ, 2019. С. 18–24. EDN: BWUPZC.

9. Федеральная рабочая программа среднего общего образования. Информатика (углубленный уровень) (для 10–11 классов образовательных организаций). https://edsoo.ru/wp-content/uploads/2023/08/22_ФРП_Информатика-10-11-классы_угл.pdf

10. Федотова В. С. Технологии имитационного моделирования в системе AnyLogic // XVII Царскосельские чтения. Материалы международной научной конференции (Санкт-Петербург, 23–24 апреля 2013 года). Т. 3. СПб.: ЛГУ имени А. С. Пушкина, 2013. С. 146–151. EDN: SBSYVH.

11. AnyLogic — имитационное моделирование для бизнеса. <http://www.anylogic.ru/>