

DOI: 10.32517/2221-1993-2026-25-3-51-61

**Н. Н. Самылкина, В. В. Рошупкина***Московский педагогический государственный университет, г. Москва, Россия*

РАЗРАБОТКА И ИССЛЕДОВАНИЕ ДИНАМИЧЕСКИХ МОДЕЛЕЙ В СРЕДЕ ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ ANYLOGIC

Аннотация

В статье рассматривается использование сред имитационного моделирования для исследования динамических процессов, происходящих в различных системах. Раскрываются возможности сред имитационного моделирования для реализации проектной работы старшеклассников. Описаны методы имитационного моделирования и примеры простых способов определения того, в каких случаях какой метод лучше применить. Приводится универсальная последовательность разработки имитационной модели. На примере решения социально значимой проблемы безопасного размещения остановки общественного транспорта рядом со школой описывается пошаговая разработка имитационной модели и ее использование для решения проблемы.

Освоение сред имитационного моделирования актуально для проведения проектных и исследовательских работ старшеклассников в связи с обновлением предметных результатов по информатике в ФГОС среднего общего образования базового и углубленного уровней изучения предмета.

Ключевые слова: ФГОС СОО, имитационное моделирование, динамические модели, проект, практико-ориентированный результат, виды имитационных моделей, среда Anylogic, уровни абстракции.

Информация об авторах

Самылкина Надежда Николаевна, доктор пед. наук, доцент, профессор кафедры теории и методики обучения математике и информатике, Институт математики и информатики, Московский педагогический государственный университет, г. Москва, Россия; *e-mail*: nsamylkina@yandex.ru

Рошупкина Виктория Витальевна, аспирант кафедры теории и методики обучения математике и информатике, Институт математики и информатики, Московский педагогический государственный университет, г. Москва, Россия; *e-mail*: vv_roshchupkina@student.mpgu.edu

1. Актуальность использования имитационного моделирования

В соответствии с требованиями Федерального государственного образовательного стандарта среднего общего образования (ФГОС СОО) учащиеся X—XI клас-

сов участвуют в проектной и исследовательской деятельности, «направленной на решение научной, личностно и (или) социально значимой проблемы», с применением «компьютерных вычислений и средств искусственного интеллекта для достижения практико-ориентированных результатов образования» [7]. Поэтому с 2027 года

N. N. Samylkina, V. V. Roshchupkina

Moscow Pedagogical State University, Moscow, Russia

DEVELOPMENT AND RESEARCH OF DYNAMIC MODELS IN THE ANYLOGIC SIMULATION MODELING ENVIRONMENT

Abstract

The article discusses the use of simulation modeling environments for studying dynamic processes occurring in various systems. It reveals the possibilities of simulation modeling environments for implementing project work by high school students. Simulation modeling methods are described, along with examples of simple ways to determine which method is best to use in different cases. A universal sequence for developing a simulation model is provided. Using the example of solving the socially significant problem of safely locating a public transport stop near a school, the step-by-step development of a simulation model and its use to solve the problem is described.

Mastering simulation modeling tools is relevant for high school students' project and research work in connection with the update of subject results in informatics in the Federal State Educational Standard of Secondary General Education for basic and advanced levels of study.

Keywords: Federal State Educational Standard of Secondary General Education, simulation modeling, dynamic models, project, practice-oriented result, types of simulation models, Anylogic environment, levels of abstraction.

Information about the authors

Nadezhda N. Samylkina, Doctor of Sciences (Education), Docent, Professor at the Department of Theory and Methods of Teaching Mathematics and Informatics, Institute of Mathematics and Informatics, Moscow Pedagogical State University, Moscow, Russia; *e-mail*: nsamylkina@yandex.ru

Viktoria V. Roshchupkina, a postgraduate student, Department of Theory and Methods of Teaching Mathematics and Informatics, Institute of Mathematics and Informatics, Moscow Pedagogical State University, Moscow, Russia; *e-mail*: vv_roshchupkina@student.mpgu.edu

в учебном предмете «Информатика» предусмотрено расширение цифровых инструментов, позволяющих моделировать и исследовать динамические процессы. В предметных результатах по информатике ФГОС СОО появились умения [там же]:

- применять программные средства на основе искусственного интеллекта для решения практических задач;
- разрабатывать и исследовать несложные имитационные модели, в частности агентные, дискретно-событийные, системно-динамические в выбранной среде моделирования;
- планировать и осуществлять учебные исследования и эксперименты с моделями для решения задач прогнозирования, поиска и оптимизации решений.

Наряду с возможностями обучения моделей нейронных сетей набирают популярность среды имитационного моделирования, которые можно использовать для решения исследовательских и проектных задач, обладая минимальными знаниями программирования.

Существуют отечественные разработки сред имитационного моделирования, например, Actor Pilgrim (МЭИ), Visual Imitak, Imitak Project (ГУУ), GPSS STUDIO (ООО «ЭЛИНА — КОМПЬЮТЕР») [6]. У программ AnyDynamics и AnyLogic существуют доступные без ограничений образовательные версии, которые можно использовать для практических работ и совместных с вузами исследовательских проектов.

Данная тема уже была реализована в учебниках для старшей школы из Федерального перечня учебников начиная с 2012 года в виде теории и практических работ [2, 3]. Более того, понятийный ряд, используемый при реализации методов имитационного моделирования, и полученные практические навыки коррелируют с тематикой искусственного интеллекта и вносят вклад в подготовку к профилю «Искусственный интеллект» Всероссийской олимпиады школьников по информатике [1, 5].

2. Возможности сред имитационного моделирования для реализации проектов

В качестве инструмента для решения практико-ориентированных задач при реализации проектов предлагается использовать **имитационное моделирование**, так как оно **позволяет**:

- проектировать системы, с анализом которых не справляются другие методы научного познания;
- анимировать моделируемую систему в режиме реального времени;
- анализировать влияние каждого элемента системы на результат эксперимента;
- прогнозировать поведение модели.

Изучение имитационного моделирования в курсе школьной информатики дает возможность моделировать учебные ситуации, приближенные к предпрофессиональным задачам, таким как анализ бизнес-процессов, работа крупных технологических комплексов или экологических систем. Также это одно из возможных профориентационных направлений, так как школьники могут перейти от теории к практике и «примерить» на себя роли предпри-

нимателя, разработчика, маркетолога, аналитика. В процессе разработки проекта учащиеся развивают такие критически важные навыки, как умение работать в команде, принимать решения и нести за них ответственность и др.

В перспективе умение создавать имитационные модели и работать с ними станет ключевым навыком для различных отраслей в связи с тенденциями цифровизации, работы с большими данными, необходимостью принимать решения в сложных и неопределенных ситуациях. Имитационное моделирование будет востребовано в управлении системами и инфраструктурой, в инженерии и проектировании, в аналитике, здравоохранении и биотехнологиях в связи с необходимостью комплексно анализировать динамические процессы, прогнозировать развитие событий и искать оптимальные решения [4, 10].

Области применения имитационного моделирования очень разнообразны, так как оно способно реализовывать различные уровни абстракции: высокий (стратегический уровень), средний (тактический уровень), низкий (оперативный уровень).

В соответствии с уровнем абстракции можно привести различные сферы, в которых активно используются имитационные модели:

- *К высокому уровню абстракции* можно отнести такие сферы, как экономика, социальные системы, экосистемы и др. В таких моделях используются более абстрактные и обобщенные понятия.
- *К моделям низкого уровня абстракции* можно отнести системы контроля, движение пешеходов, движение транспортных средств и др. Эти процессы и объекты моделируются максимально точно. В них необходимо учитывать различные физические характеристики, которые могут повлиять на протекание процесса (ширина, длина, высота, расстояние, скорость и т. д.).
- *К моделям среднего уровня абстракции* относятся бизнес-процессы, работа складов, больниц, аэропортов, мультимодальных терминалов. Эти модели сочетают в себе характеристики и высокого, и низкого уровней абстракции [8].

Разработчик имитационной модели самостоятельно выбирает уровень абстракции. От выбора уровня абстракции зависят эффективность и корректность модели. Зачастую уровень абстракции может быть изменен непосредственно в процессе разработки модели.

3. Виды имитационных моделей

В теоретическую подготовку учащихся перед применением имитационных моделей входит рассмотрение методов имитационного моделирования. Поскольку теория должна быть адаптирована для школьников и сложная математика здесь не будет использоваться, можно вместо методов говорить о **видах имитационного моделирования**:

- агентное;
- дискретно-событийное;
- системная динамика.

Агентное моделирование позволяет создавать модели различных уровней абстракции — от низкого до высокого. С помощью него можно рассмотреть взаимодействие отдельных и независимых друг от друга объектов (агентов),

каждый из которых имеет следующие свойства: установленные правила поведения, механизм принятия решений в различных ситуациях. К этому виду относится, например, моделирование движения пешеходов на перекрестке.

В рамках **дискретно-событийного моделирования** разрабатываются модели среднего или низкого уровня абстракции, позволяющие изучать поведение системы, представленной набором последовательных событий. Это, например, работа конвейера на фабрике — любой продукт проходит определенные этапы создания в строго заданном порядке.

Метод системной динамики используется для моделей с высоким уровнем абстракции, зачастую он применяется для прогностических моделей. Системно-динамический метод моделирования служит для изучения поведения систем, усложненных большим количеством обратных связей и зависимых параметров. Примерами такой системы являются крупный город, большое производство, демографические процессы [9].

Старшеклассники самостоятельно смогут определить вид (метод) моделирования в зависимости от поставленных целей и задач.

На этапе знакомства с имитационным моделированием **школьникам могут быть предложены условные формулы определения вида (метода) моделирования** (рис. 1).

*Агенты + правила = **агентное моделирование***

*Агенты + случайные события + очереди =
= **дискретно-событийное моделирование***

*Зависимости между параметрами без явных
агентов = **системная динамика***

Рис. 1. Условные формулы определения вида (метода) моделирования

Прежде чем старшеклассники начнут моделировать различные процессы, нужно, чтобы они умели определять:

- проблемные ситуации;
- моделируемые процессы;
- параметры этих процессов;
- используемый метод имитационного моделирования.

Для этого можно использовать карточки с изображением моделей (рис. 2–4) — учащиеся, глядя на них, определяют вышеперечисленные компоненты.

Карточка 1: Эпидемия.

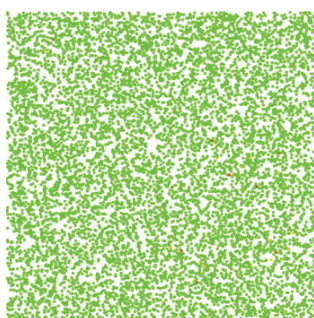


Рис. 2. Карточка 1: Эпидемия (агентное моделирование)

Проблемная ситуация: эпидемия.

Процесс: заражение агентов.

Параметры:

- продолжительность болезни;
- расстояние между агентами;
- инкубационный период.

Карточка 2: Лесной пожар.

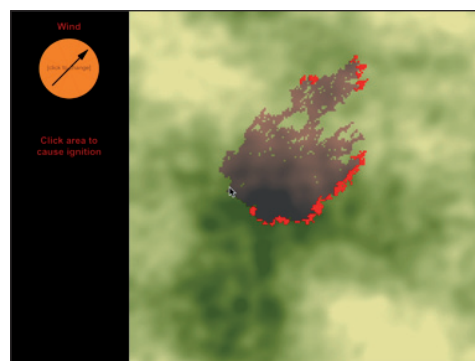


Рис. 3. Карточка 2: Лесной пожар (системная динамика)

Проблемная ситуация: лесной пожар.

Процесс: увеличение площади возгорания.

Параметры:

- место возгорания;
- направление ветра;
- скорость ветра.

Карточка 3: Склад.

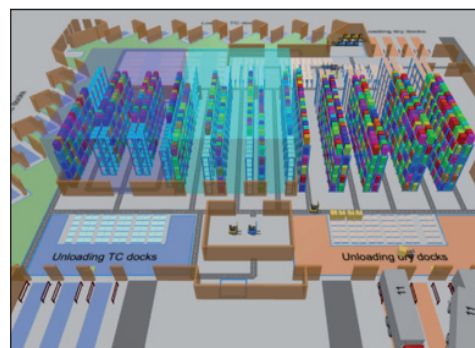


Рис. 4. Карточка 3: Склад (дискретно-событийное моделирование)

Проблемная ситуация: логистика скоропортящихся продуктов.

Процессы:

- получение товара;
- сортировка товара;
- хранение товара;
- получение заказов;
- сбор заказов;
- подготовка распределительных линий;
- погрузка заказов;
- доставка заказов.

Параметры:

- тип товара;
- габариты и вес товара;
- срок годности товара;
- требования к условиям хранения, продажи и др.

Здесь же можно предложить учащимся **универсальную последовательность действий для создания модели процесса в среде имитационного моделирования**:

- первый шаг заключается в выборе предметной области или проблемной задачи;
- на втором шаге требуется определить процесс моделирования, от которого зависит решение задачи;
- затем следует определить параметры процесса, метод имитационного моделирования, например, объекты, продолжительность, цикличность;
- следующий шаг — непосредственное создание модели;
- и завершающий этап — анализ результатов работы, проверка корректности.

4. Разработка имитационной модели по шагам

Приведем пример практической работы, которую учащиеся могут выполнить на уроке.

Выявление социальной проблемы:

Учащиеся по приезде на остановку вынуждены добираться до школы через нерегулируемый участок дороги, на котором отсутствует тротуар и имеется парковка, вследствие чего доступ учеников к школе от места остановки общественного транспорта не является безопасным [8].

Проблема: потенциально небезопасное расположение остановки общественного транспорта.

Цели: найти безопасный вариант расположения остановки общественного транспорта и протестировать его в среде имитационного моделирования.

Задачи:

- проанализировать причины небезопасного доступа к школе;
- изучить нормативы, которые должны соблюдаться при установке социальных объектов;
- разработать варианты разрешения проблемы;
- создать модель и проанализировать эффективность предложенных вариантов;
- подготовить обращение к ГИБДД;
- подготовить обращение к управе района Северное Медведково для переноса остановки общественного транспорта.

В процессе разработки проекта были изучены стандарты ГОСТ Р 52766-2007 «Дороги автомобильные общего пользования. Элементы обустройства. Общие требования» и ОСТ 218.1.002-2003 «Автобусные остановки на автомобильных дорогах», в соответствии с которыми разрабатывался вариант решения поставленной проблемы.

Создание модели включает следующие этапы:

- Создание дороги.
- Редактирование перекрестка.
- Создание потоков движения транспортных средств.

- Создание парковки.
- Создание автобусных остановок.
- Создание потоков движения автобусов.
- Запуск школьного транспорта.
- Создание пешеходов.

Для реализации модели была выбрана среда имитационного моделирования AnyLogic. Данная программа имеет ряд преимуществ в сравнении с другими программными средствами:

- наличие бесплатной образовательной версии;
- доступный интерфейс на русском языке;
- возможность использования нескольких видов (методов) моделирования: дискретно-событийного, системно-динамического и агентного;
- наличие справочных и обучающих материалов в открытом доступе.

Для более эффективного обучения работе с имитационными моделями в AnyLogic учащимся следует предлагать оптимизационные задачи, сюжет которых основан на наглядной и понятной ученикам проблеме.

Помимо возможности создавать наглядные модели учащиеся смогут глубже освоить язык программирования Java, если ранее были с ним знакомы. Созданная модель в среде AnyLogic автоматически отображается на Java-коде. Особо важен язык программирования в создании более сложных моделей, моделей реального мира, в которых зачастую требуется использование функции вероятностных распределений для вычисления выражений, описания свойств объектов, а также их связей. Использование языка программирования Java позволяет сделать модели кроссплатформенными, т. е. они будут поддерживаться в любой Java-среде.

Перейдем к созданию модели в AnyLogic. Откроем среду на начальной странице.

Создадим проект с названием «Transport System» и модельными единицами времени — секундами.

Для вставки изображения в рабочую область следует на панели *Палитра* перейти в раздел *Презентация* и выбрать *Изображение*, перетащив его левой кнопкой мыши в рабочую область.

Затем необходимо масштабировать изображение, указав в его свойствах пропорционально уменьшенные или увеличенные высоту и ширину. В данной модели разрешение изображения составило 750×492 пикселя.

Для построения транспортной модели будем использовать изображение интернет-карты (рис. 5), где указаны пешеходный переход, текущая автобусная остановка, потенциально опасная зона, парковка и место расположения будущей остановки.

Чтобы избежать случайного изменения масштаба или перемещения карты во время построения модели, требуется заблокировать объект. Для этого в свойствах изображения используется пункт *Блокировать*.

Для корректного расчета расстояния между объектами и времени преодоления расстояния между ними нужно применить *Линейку*, соотнеся ее с линейкой карты, находящейся в правом нижнем углу, в свойствах указать ее размер в соответствии с масштабом на карте — 10 метров.

На панели *Палитра* в разделе *Библиотека дорожного движения* двойным щелчком мыши следует



Рис. 5. Интернет-карта

выбрать объект *Дорога*. Откладывать дорогу на карте нужно по разделительной полосе с конца дороги по направлению основного движения. По направлению дороги следует нажать несколько раз на разделительную полосу, в конце дороги сделать двойной щелчок мышью. Один щелчок мышью создает узел дороги, который в дальнейшем поможет сделать дорогу более извилистой, максимально приближенной к реальной. Двойной щелчок позволяет обозначить конец дороги.

Аналогичным образом создается вторая дорога, но ее конец должен лежать на разделительной полосе другой дороги. Таким образом, автоматически создается перекресток (рис. 6). Также для создания перекрестка можно воспользоваться инструментом *Перекресток* из *Библиотеки дорожного движения*. После создания перекрестка в модели основная дорога будет разбита уже на две дороги.

В свойствах дороги должны быть указаны ее соответствующие характеристики (рис. 7). В данной модели используются главная двусторонняя дорога с двумя полосами движения в каждом направлении и второстепенная двусторонняя дорога с одной полосой движения в каждом направлении.

Программа автоматически создаст на перекрестке возможные пути передвижения. Важно их настроить в соответствии с реальным движением транспорта для корректного исследования и избежания возникновения ошибок. Для создания или удаления направления достаточно один раз нажать на него. В данной модели



Рис. 6. Дорога

road - Дорога

Имя:

☐ Исключить ☒ Отображается на верхнем агенте

☐ Блокировать

Видимость: ☒ да

☐ Одностороннее движение

Кол-во полос основного движения:

Кол-во полос встречного движения:

Ширина разделительной полосы: м

Цвет разделительной полосы:

road2 - Дорога

Имя:

☐ Исключить ☒ Отображается на верхнем агенте

☐ Блокировать

Видимость: ☒ да

☐ Одностороннее движение

Кол-во полос основного движения:

Кол-во полос встречного движения:

Ширина разделительной полосы: м

Цвет разделительной полосы:

Рис. 7. Свойства дорог

перекресток имеет направления движения, изображенные на рисунке 8.

В качестве тротуара в данной модели использовалась *Ломаная* из раздела *Презентация*. Необходимо увеличить ее толщину до 10 pt и изменить координаты (29, 230, -10) для корректного отображения слоев (тротуар находится поверх карты). В свойствах объекта следует установить светло-серый цвет для наглядного различия дороги и тротуара (рис. 9).

При помощи кнопки *Скрыть элементы разметки пространства* можно сделать полупрозрачными элементы разметки и посмотреть, насколько построенная



Рис. 8. Направления движения на перекрестке

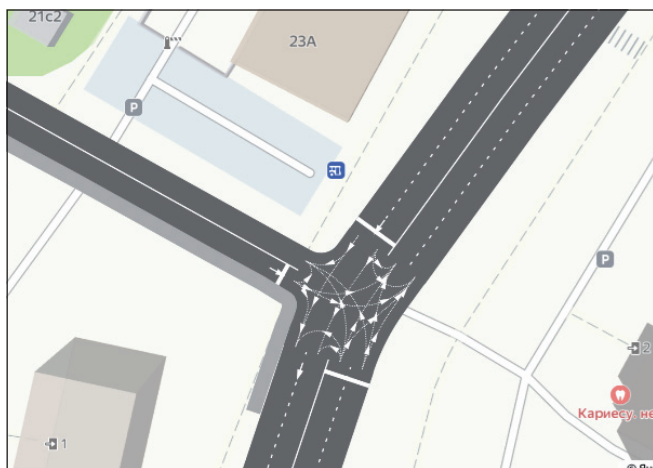


Рис. 9. Тротуар

дорожная сеть соответствует спутниковому снимку или карте, а в случае некорректного построения изменить расположение дорог.

Если у агента можно выделить несколько состояний, выполняющих различные действия при происхождении каких-то событий, или если у агента есть несколько качественно различных поведений, последовательно сменяющих друг друга при происхождении определенных событий, то поведение такого объекта может быть описано в терминах **диаграммы состояний** [7]. Используя диаграмму состояний, можно графически задать множество различных состояний объекта и действия, происходящие при их смене, а также задать события, являющиеся причинами переходов агентов модели из одних состояний в другие.

Для создания диаграммы состояний были использованы следующие основные блоки библиотеки дорожного движения:

- **carSource**. Используется для создания потока машин. Для работы данного блока необходимо в его свойствах указать дорогу, на которую будут помещены автомобили при их создании.
- **carMoveTo**. Данный блок отвечает за конечный пункт прибытия объекта модели. Определение конечного пункта объекта непосредственно зависит от целей моделирования. Для работы данного блока необходимо в его свойствах указать дорогу или парковку, автобусную остановку, стоп-линию, на которую в дальнейшем должен быть помещен автомобиль. AnyLogic самостоятельно определяет путь движения от исходной дороги до пункта назначения.
- **carDispose**. После достижения транспортом поставленной цели в блоке **carMoveTo** машина удаляется из модели при помощи данного блока.

Перенесем вышеперечисленные блоки в рабочую область и соединим их, как показано на рисунке 10. В свойствах первого укажем дорогу **road**, второго — **road1**.

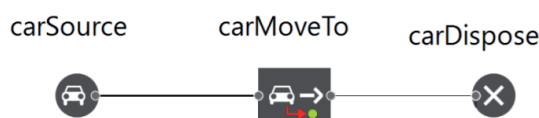


Рис. 10. Диаграмма состояний автомобилей

Помимо дорог в свойствах блока **carSource** необходимо указать интенсивность прибытия транспортных средств в час, направление движения и характеристики, такие как длина, скорость, ускорение и торможение автомобилей. В свойствах блока **carMoveTo** — цель и направление движения (рис. 11).

Если на данном этапе запустить модель, то откроется новое диалоговое окно, демонстрирующее построенную имитационную модель, в которой вместо машин изображены движущиеся прямоугольники (рис. 12).

Для того чтобы прямоугольники приняли вид автомобилей, требуется создать новый тип агента. Для этого в свойствах блока **carSource** в разделе **Автомобиль** нужно выбрать **Создать другой тип**, в появившемся диалоговом окне указать имя **Car**, выбрать **Далее**, в следующем шаге указать анимацию **Автомобиль** и нажать **Готово**. После выполнения ранее перечисленных действий при запуске модели можно увидеть автомобили вместо прямоугольников (рис. 13).

Аналогичным образом в модель были добавлены другие потоки (табл. 1).

Вдоль дороги требуется установить параллельную парковку для автомобилей. Для этого из **Библиотеки дорожного движения** следует воспользоваться инстру-

carSource - CarSource

Имя: ☒ Отображать имя

☐ Исключить

Прибывают согласно:

Интенсивность прибытия: в час

Считать параметры агентов из БД: ☐

Ограниченное кол-во прибытий: ☐

Появляется: ☒ на дороге ☐ на парковке

Дорога:

Помещается на полосу: ☒ основного движения ☐ встречного движения

Случайная полоса: ☒

carMoveTo - CarMoveTo

Имя: ☒ Отображать имя

☐ Исключить

Цель движения: ☒ дорога ☐ парковка ☐ стоп-линия ☐ автобусная остановка

Дорога:

Доехать до конца полосы: ☒ основного движения ☐ встречного движения

Рис. 11. Свойства блоков **carSource** и **carMoveTo**

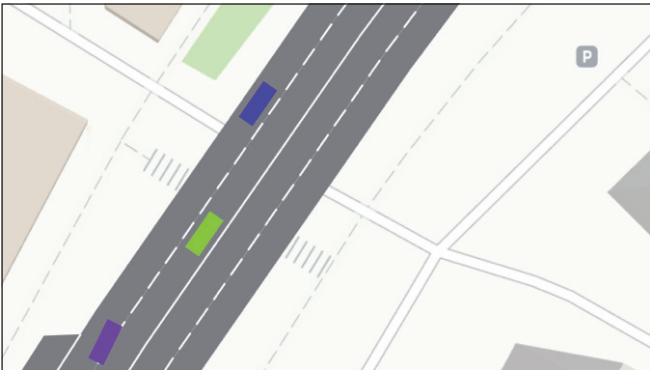


Рис. 12. Графическое представление автомобилей

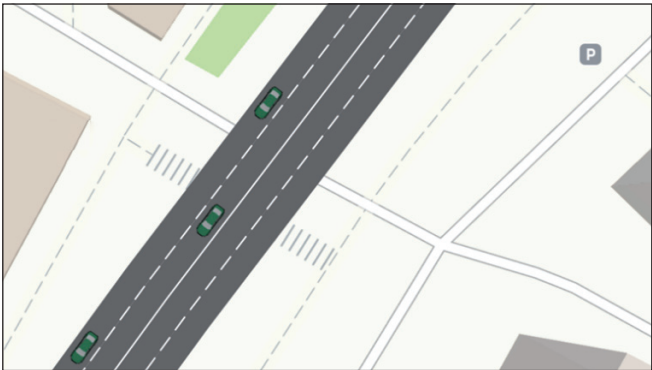


Рис. 13. Представление автомобилей

Таблица 1

Направления движения автомобилей

Пункт отправления	Пункт прибытия	Интенсивность (в единицах выбранного модельного времени)	Полоса отправления	Полоса прибытия
road1	road	450	Встречная	Встречная
road1	road2	150	Встречная	Встречная

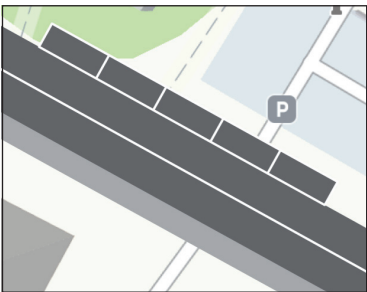


Рис. 14. Парковка

ментом *Парковка*. Левой кнопкой мыши нужно перенести парковку в рабочую область (рис. 14).

В ранее созданную диаграмму состояний необходимо добавить дополнительные блоки. Из *Библиотеки моделирования процессов* воспользуемся блоком *selectOutput*, который позволяет с заданной вероятностью разделить автомобили на два потока (часть автомобилей останавливается на парковке, остальные продолжают движение). Также потребуется два блока *carMoveTo5* и *carMoveTo7*, целью движения первого блока является парковка, второго — дорога, на которой автомобиль продолжает свое движение. Блок ожидания *Delay2* позволяет автомобилям задерживаться на парковке (рис. 15).

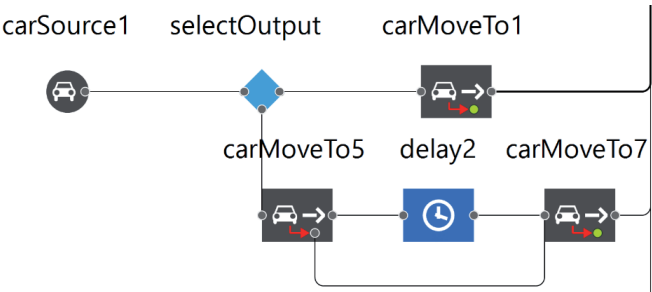


Рис. 15. Диаграмма состояний автомобилей. Парковка

На следующем этапе требуется создать автобусную остановку. Для этого необходимо воспользоваться инструментом *Автобусная остановка* из раздела *Библиотека дорожного движения* и разместить ее на месте, которое было определено в качестве альтернативы к текущему положению остановки (рис. 16).

Для автобусов нужно создать диаграмму состояний. Она состоит из тех же блоков, что и диаграмма состояний для автомобилей и блока ожидания *Delay*. Этот блок задерживает автобусы на некоторое время на остановке, пока из него выходят пассажиры. Чтобы задать изображения автобусов, потребуется создать новый тип агента в разделе *Автомобиль*. Этапы создания нового типа

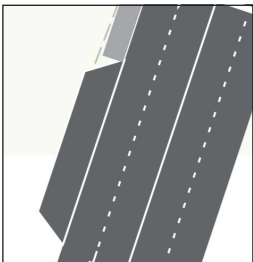


Рис. 16. Автобусная остановка

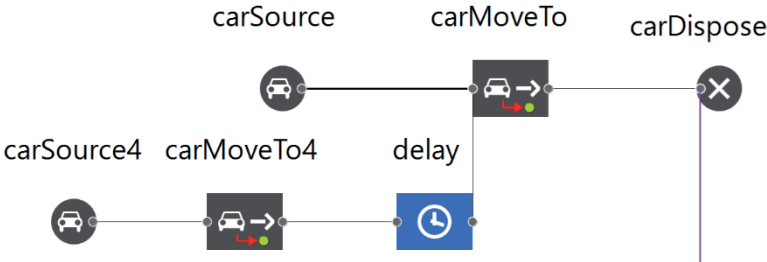


Рис. 17. Диаграмма состояний автобусов

Таблица 2

Направления движения автобусов

Пункт отправления	Пункт прибытия	Пункт прибытия	Интенсивность	Полоса отправления	Полоса прибытия
road	busStop	road1	Согласно расписанию	Основная	Основная
road1	busStop1	road	Согласно расписанию	Встречная	Встречная

Свойства

schedule - Расписание

Имя: ☒ Отображать имя ☐ Исключить

Видимость: ☒ да

Данные

Тип:

Расписание задает: ☒ Интервалы (Начало, Конец) ☐ Моменты времени

Длительность: ☐ Неделя ☐ Дни/Недели ☒ Другая (нет привязки к календарю)

Повторять каждые: мин.

☐ Привязать к мин.

Значение по умолчанию:

☐ Загружается из базы данных

Начало	Конец	Значение
0	5	1

Рис. 18. Расписание автобусов

агентов аналогичны ранее описанным (см. рис. 15). Надо сделать так же и для автобусов (как показано на рис. 17). В качестве названия было выбрано имя *Bus* и анимация изображения *Автобус1*.

В таблице 2 указано направление движения автобусов в созданной модели.

У каждого маршрута существует установленное расписание движения, которое необходимо отразить в данной модели, поэтому в разделе *Библиотека моделирования процессов* используется инструмент *Расписание*. Согласно расписанию, автобусы прибывают каждые пять минут, это следует отразить в свойствах объекта.

Тип данных расписания является целым, так как количество прибывающих автобусов не может быть вещественным числом. Расписание будет задавать интервалы времени для точного нормирования прибытий автобусов. Когда интервалы времени в таблице заканчиваются, они повторяются с цикличностью 5 мин, это свойство отражено в разделе *Повторять каждые* (рис. 18).

При помощи инструментов *Стена* и *targetLine* из *Пешеходной библиотеки* нужно ограничить область движения будущих пешеходов и указать пункты отправления и прибытия. Для этого необходимо установить целевые линии возле автобусной остановки и перед

школой, по периметру тротуара возвести стену, конец и начало которой совпадают (рис. 19). В свойствах стены нужно убрать галочку из поля *Отображается на верхнем агенте*.

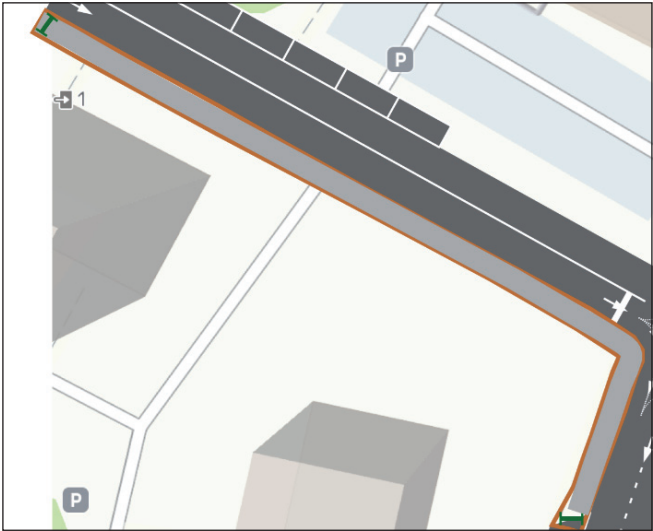


Рис. 19. Стена, целевые линии движения пешеходов



Рис. 20. Диаграмма состояний пешеходов

Свойства [иконка]

schedule1 - Расписание

Имя: ☒ Отображать имя ☐ Исключить

Видимость: ☒ да

Данные

Тип:

Расписание задает: ☒ Интервалы (Начало, Конец) ☐ Моменты времени

Длительность: ☐ Неделя ☐ Дни/Недели ☒ Другая (нет привязки к календарю)

Повторять каждые: сек.

☐ Привязать к сек.

Значение по умолчанию:

☐ Загружается из базы данных

Начало	Конец	Значение
29	329	20
329	629	20
629	929	25
929	1229	30

[+] [↑] [↓] [X]

Рис. 21. Расписание пешеходов

Для создания учащихся, выходящих из автобуса и направляющихся к школе, используется *Пешеходная библиотека*. Диаграмма состояний пешеходов создается при помощи блоков *PedSource*, *PedGoTo* и *PedSink* (рис. 20).

В свойствах блока *PedSource* необходимо указать *TargetLine*, в *PedGoTo* — *TargetLine2*, что обозначает начало и конец пути пешеходов. Чтобы агенты пешеходов имели образ человека, требуется создать новый тип *Pedestrian* с анимацией *Человек*.

Время прибытия пешеходов неразрывно связано со временем прибытия автобусов. Поэтому время прибытия пешеходов также можно задать при помощи инструмента *Расписание*. Важно отметить, что в данном случае в расписании в качестве единиц времени используются се-

кунды, так как на прибытие каждого автобуса отводится 5 мин, 29 сек из которых затрачивается на движение от начала пути в модели до остановки. Так как количество учащихся, приезжающих на автобусах, зависит от времени прибытия (ближе к началу урока количество учащихся увеличивается), то в расписании требуется указать все интервалы времени с соответствующим значением прибывающих школьников (рис. 21).

Для удобного просмотра построенной модели предлагается из раздела *Презентация* воспользоваться инструментом *3D окно*. Его рекомендуется разместить рядом с основным окном просмотра модели. Результат просмотра в трехмерном режиме представлен на рисунке 22. Чтобы настроить ракурс показа модели,

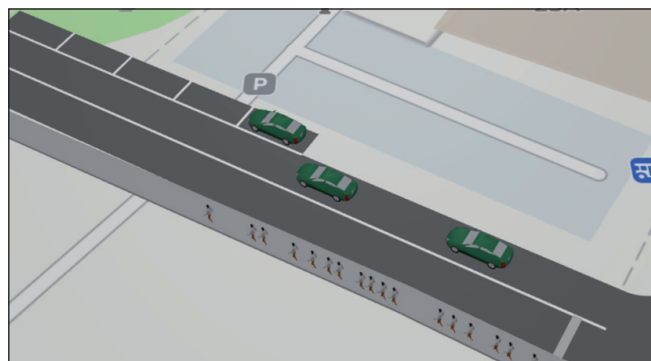
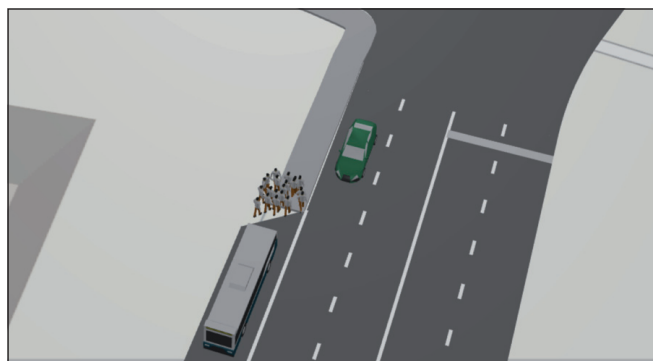


Рис. 22. Трехмерная модель

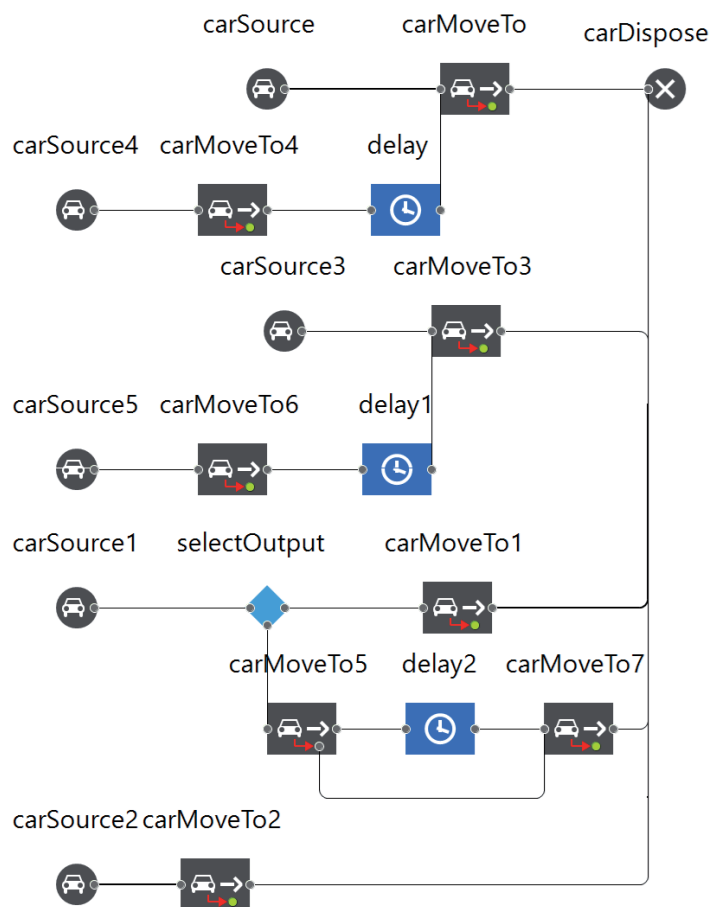


Рис. 23. Диаграмма состояний всей модели

необходимо воспользоваться инструментом *Камера*, установив ее на изображении карты в требуемом месте и указав в свойствах нужный угол поворота.

На данном этапе модель готова. В модели были учтены расписание прибытия автобусов, интенсивность прибытия школьников в зависимости от времени, другие участники движения. Получившаяся диаграмма состояний представлена на рисунке 23.

Режим просмотра готовой модели в двумерной и трехмерной проекциях изображен на рисунке 24.

В ходе работы над проектом учащиеся должны рассмотреть различные варианты переноса остановки, в результате анализа которых в среде имитационного моделирования AnyLogic будет выбран оптимальный вариант.

После оглашения учениками найденных решений можно продемонстрировать один из возможных вариантов решения данной проблемы (см. рис. 24), обсудить его вместе со школьниками, а также сравнить с другими результатами.

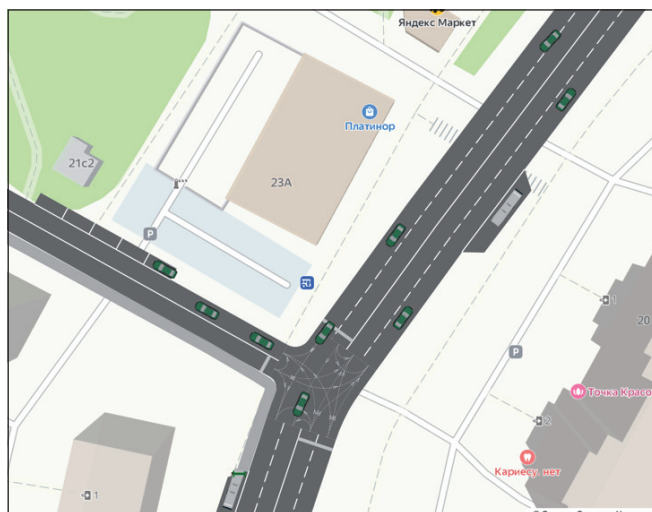


Рис. 24. Модель автобусной остановки

5. Заключение

Использование сред имитационного моделирования может стать практическим инструментом, позволяющим проводить виртуальные эксперименты и проверять гипотезы в рамках проектной деятельности. Имитационное моделирование предоставляет возможность изменять параметры процессов в режиме реального времени и наблюдать последствия принимаемых решений, что способствует формированию у школьников понимания причинно-следственных связей, формирует культуру исследовательской деятельности.

Финансирование

Научная статья подготовлена в рамках Государственного задания Министерства просвещения Российской Федерации на 2026 год ФГБНУ «Институт содержания и методов обучения имени В. С. Леднева» по проекту «Научно-методические основы преемственности естественно-научного и математического образования на уровне среднего общего образования».

Список источников

1. Информационно-методическое письмо об особенностях преподавания учебного предмета «Информатика» в 2025/26 учебном году. <https://edsoo.ru/wp-content/uploads/2025/08/informatika.pdf>

2. Калинин И. А., Самылкина Н. Н. Информатика. Углубленный уровень. 10 класс: учебник. М.: БИНОМ. Лаборатория знаний, 2013. 256 с.

3. Калинин И. А., Самылкина Н. Н., Бочаров П. В. Информатика. Углубленный уровень: задачник-практикум для 10–11 классов. М.: БИНОМ. Лаборатория знаний, 2014. 248 с.

4. Карнов Ю. Г. Имитационное моделирование систем. Введение в моделирование с Anylogic 5. СПб.: БХВ-Петербург, 2009. 400 с.

5. Комплексный план мероприятий по повышению качества математического и естественно-научного образования на период до 2030 года (утв. распоряжением Правительства Российской Федерации от 19 ноября 2024 года № 3333-р). <http://government.ru/docs/53427/>

6. Малыгина С. Н., Неупокоева Е. О. Обзор современных средств имитационного моделирования // Труды Кольского научного центра РАН. Серия: Технические науки. 2022. Т. 13. № 2. С. 134–143. DOI: 10.37614/2949-1215.2022.13.2.013.

7. Проект приказа Минпросвещения России «Об утверждении федерального государственного образовательного стандарта среднего общего образования». <https://docs.edu.gov.ru/document/95cddc2389fb11bdaf9cac0221b4fc45>

8. Рощупкина В. В. Формирование компетенций школьников посредством индивидуального и группового стартапов // Актуальные проблемы методики обучения информатике и математике в современной школе. Материалы Международной научно-практической интернет-конференции. Памяти академика РАО, заслуженного учителя РФ Людмилы Леонидовны Босовой (Москва, 22–30 апреля 2025 года). М.: Московский педагогический государственный университет, 2025. С. 273–278. EDN: VSBCBP.

9. Хроль Е. В., Уварова А. Г., Кужильный А. В. Разработка имитационных моделей с помощью AnyLogic // Современные инновации, системы и технологии. 2023. Т. 3. № 4. С. 119–130. DOI: 10.47813/2782-2818-2023-3-4-0119-0130. EDN: UYVFEM.

10. Юрченко Т. В. Информационные технологии в экономике. Решение экономических задач средствами MS EXCEL 2007: учебное пособие. Н.Новгород: Нижегородский государственный архитектурно-строительный университет, 2010. 132 с.