

## МОДЕЛИРОВАНИЕ ПОВЕДЕНИЯ ТОЛПЫ ЛЮДЕЙ НА БАЗЕ КЛЕТОЧНЫХ АВТОМАТОВ

**Плотников Д.Ю., Аноприенко А.Я.**

Донецкий национальный технический университет г.Донецк

Кафедра компьютерной инженерии

E-mail: [dayko1@ukr.net](mailto:dayko1@ukr.net)

### **Аннотация**

*Плотников Д.Ю., Аноприенко А.Я. Моделирование поведения толпы людей на базе клеточных автоматов. В статье исследуется вопрос использования клеточных автоматов для моделирования поведения толпы людей. В роли людей выступают активные клетки. Также в программе есть элемент «преграда», при помощи которого моделируется окружающая среда. Существует несколько наборов правил поведения людей и факторов влияния, что позволяет моделировать различные сценарии развития и движения. Поскольку в наше время массовые мероприятия проводятся каждый день, моделирование крупных потоков людей является актуальным, особенно в свете скорого проведения в Украине Евро-2012.*

### **Постановка задач исследования**

В данной работе задачей является разработка программы, ориентированной на моделирование поведения толпы людей, исследование поведения людей в различных ситуациях (на концертах, стадионах и т.д.).

### **Решение задачи и результаты исследований**

В наше время каждый день проводится множество массовых мероприятий. В одно единственное мгновение при определенном стечении событий люди могут превратиться в неуправляемую толпу, поведение которой сложно предсказать. На самом деле, даже организованная масса людей может представлять собой угрозу.

Очень часто неуправляемый поток людей приносит разрушения и жертвы, поэтому на всех массовых мероприятиях должен быть четкий план действий для нейтрализации беспорядка и паники. Для разработки такого плана необходимо промоделировать все возможные, или же наиболее вероятные, сценарии развития событий при столпотворении людей, сделать необходимые выводы и провести организационные мероприятия на местах.

Я поставил перед собой задачу промоделировать поведение толпы людей по различным правилам развития событий.

Моделирование проводится на базе клеточных автоматов. За основу взята программа «Жизнь», которая была разработана ранее. Исследования этой игры показали, что ее модификации подходят для моделирования поведения массы людей.

Программа написана в среде Borland C++ Builder 6. В дальнейшем планируется разработка аналогичной программы в качестве Web-приложения на языке Action Script 3.0.[1]

На рисунке 1 можно видеть интерфейс программы. Для начала работы программы необходимо открыть существующий проект или создать новый. В качестве фонового изображения можно задать либо пустой белый фон либо любую картинку в формате bmp с разрешением 640 на 640 точек. Следующим шагом будет нанесение «местности» на карту, а потом запустить моделирование. «Местностью» в данной программе называется совокупность объектов на поле для моделирования. Объекты могут быть следующих типов: точки, преграды, опасности.[2]

Далее рассмотрим подробнее каждый тип и примеры работы программы.

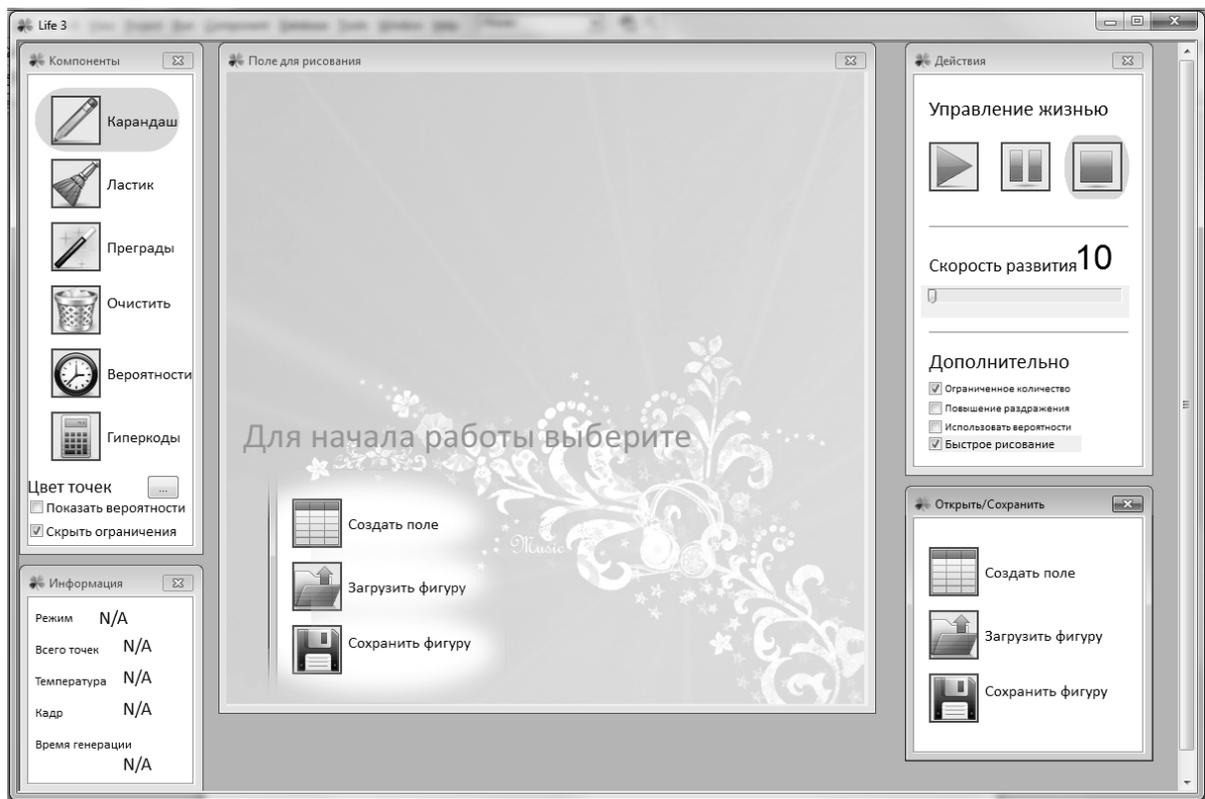


Рисунок 1 – Интерфейс программы

Тип «Точка» подразумевает под собой живого человека. Он обладает рядом параметров, а именно «Раздраженность», «Цвет». Для каждой «точки» можно задать «цвет», а «раздраженность» меняется сама, в ходе работы программы и зависит от того, сколько рядом с точкой находится соседей.

«Преграда» представляет собой ограничитель на поле, сквозь который не может прийти ни одна «точка». В нашей жизни это большой камень или столб.

«Опасности» это особый вид элементов, которые влияют по поведение клеток рядом с ними. Клетки стремятся убежать от опасности как можно дальше. Расстояние, на которое перемещаются клетки от «опасностей» зависит от степени «опасности», которую можно менять.

Для того чтобы нанести на карту различные типы объектов необходимо выбрать их слева в меню и нарисовать на поле, как в графическом редакторе. Для упрощения нанесения элементов была добавлено возможность использования гиперкодов, то есть представление любого (вплоть до размера поля) пар координат двумя словами. Рассмотрим более подробно:

Нам нужно задать 2 точки с координатами  $x1=2_{10}(10_2)$ ,  $y1=2_{10}(10_2)$   $x2=2_{10}(10_2)$ ,  $y2=0_{10}(00_2)$ . Можно заметить, что координаты точек  $y1$  и  $y2$  отличаются только в одном бите. В программе такую ситуацию мы записываем как  $0M$ . Тогда программа вместо  $M$  подставляет поочередно 0 и 1. Пример работы мы можем видеть на рисунке 2. Также гиперкоды удобно задавать в тому случае, когда местность для моделирования это часть города с параллельными улицами. Если посмотреть карту любого из городов мира то можно заметить, что таких частей очень много.

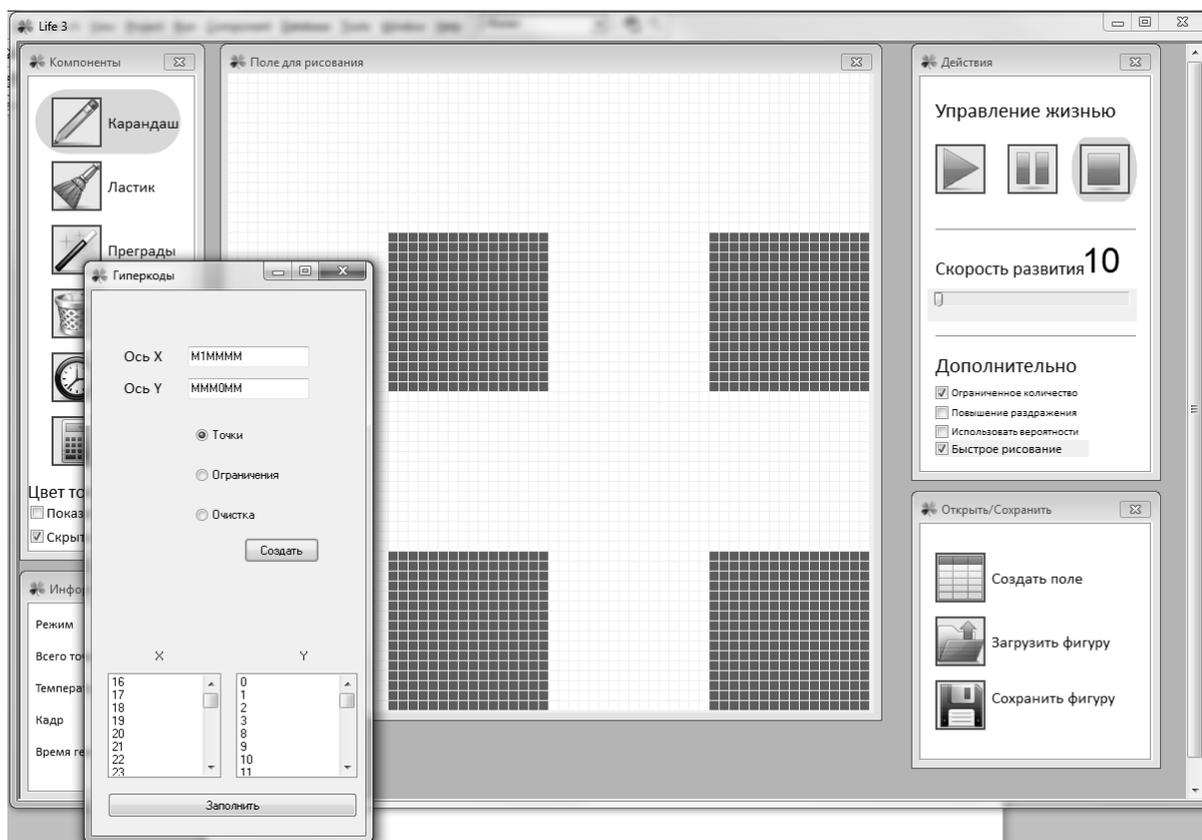


Рисунок 2 – Задание «Точек» с помощью гиперкодов

За основу алгоритма был взят алгоритм игры «Жизнь», придуманный Джоном фон Нейманом. Основная его идея заключалась в том, что анализируются соседи клетки. Если у нее больше или равно двум – она выживает. Если меньше – погибает. В алгоритме идет анализ каждой ячейки слева направо и сверху вниз, но изменения в текущем шаге не учитываются. То есть в начале каждого цикла создается пустая матрица, такого же размера как и исходная. Идет анализ каждой клетки исходной матрицы, а результат записывается в буферную матрицу. Когда последняя клетка была проанализирована – программа переписывает результат из буферной в исходную и отображает результат. И так пока не нажмут кнопку «Стоп». Из-за такой обработки можно встретить много красивых динамических фигур, но нигде в нашей жизни их мы встретить не можем. В программе же было принято решение изменить алгоритм на более реалистичный. Перед началом обработки выбирается случайное число клеток (не больше максимально допустимого), выбираются клетки и анализируются только выбранные. Это позволило избавиться от устойчивых динамических фигур (например Планер) и оставить только статистическую (квадрат). Следующей особенностью стало то, что в данном случае идет быстрый рост клеток. Его можно ограничивать тем, что выбирается на обработку число не более заданного.

Точки постоянно взаимодействуют друг с другом. Чем больше их находится рядом, тем больше растет их «возмущение». В программе реализована функция предупреждения о слишком большом возмущении точек в каком-либо районе. Если собралось несколько точек вместе в одной части программы, то разрядность в этом участке увеличивается, что позволяет снизить «раздражение» точек и не допустить критической ситуации. В жизни

можно провести аналогию с поведением людей на концерте. Бывает, что люди стоят на расстоянии 1 метр друг от друга, потом расстояние сокращается до 30, 15 см и так далее. В определенный момент будет критическая точка, т.е. места уже на всех не хватит и возникнет конфликт. В программе разрядность может увеличиваться заданное количество раз и при попытке увеличить ее больше заданного – будет выведено предупреждение, что в некоторой области возник конфликт, что в реальной жизни означает давка или паника среди народа. На рисунке 3 изображен процесс наступления критической точки.

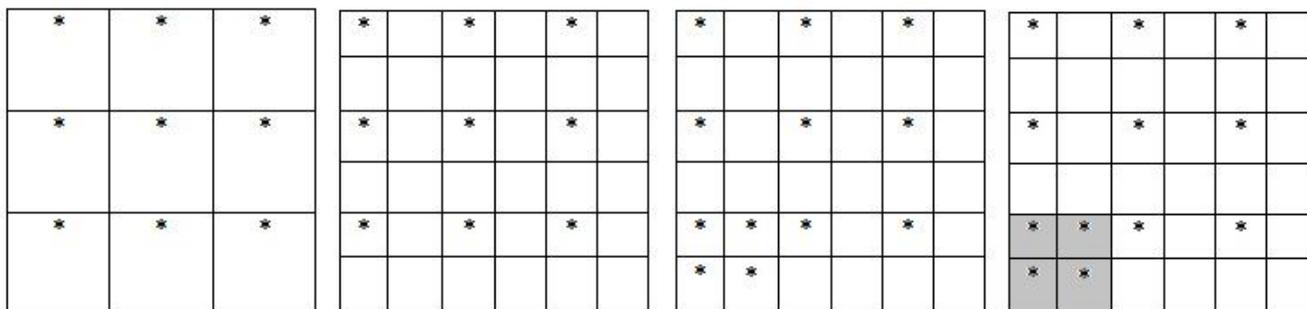


Рисунок 3 – Процесс наступления критической точки

Инструмент на панели слева «Вероятности» задает точки опасности, от которых «убегают» «точки». Уровень опасности можно задать. От 1 до 32000. Сейчас исследовано несколько опасностей, но к моменту выхода следующей версии программы будет проанализировано большинство современных угроз и выставлен им уровень. Для текущих угроз уровень опасности задан в таблице 1

Таблица 1 – Уровни опасности угроз

Название	Уровень
Пожар	5
Наводнение	5
Появление террориста с пакетом вызрывчатки	8

На рисунке 4 мы можем видеть пример работы программы. Люди находятся в определенной части города и в некоторый момент происходит событие: появление террориста среди толпы. Люди начинают убегать, но проход очень узкий и возникает давка. Появление террориста мы сами задаем, нажав во время моделирования кнопку «Пауза» и нарисовав событие запускаем снова на выполнение кнопкой «Играть». Кнопка «Стоп» служит для возврата поля к состоянию предыдущего запуска. На рисунке 4 опасность это 4 светлых клетки.



Рисунок 4 – Поведение людей при появлении опасности

## Выводы

Была создана программа, которая позволяет моделировать поведение толпы в различных ситуациях. При запущенном процессе мы видим подробную информацию о ходе работы программы. В любой момент можно приостановить процесс и добавить любой элемент на поле. Как можно было видеть на рисунке 4 мы после запуска подождали некоторое время, потом нажали паузу, добавили элемент «Опасность» и вновь запустили программу. Для того, чтобы алгоритм как можно точнее соответствовал реальной жизни были внесены элементы случайности, то есть все люди одновременно не могут двигаться а так же учтена «раздражительность». Данный параметр изменятся в зависимости от того, сколько людей находится рядом друг с другом насколько активно они взаимодействуют. Для того, чтобы можно было найти проблемные места на карте, например выбрать оптимально расположение дверей в коридорах в метро или на концерте и не допустить давки – был внесен элемент увеличения разрядности определенной клетки в заданное количество раз, и если увеличить уже невозможно – программа сигнализирует об этом.

## Список литературы

[1] Аноприенко А.Я., Плотников Д.Ю., Мальований С.Ф. Использование клеточных автоматов для моделирования движения транспорта. Информационные управляющие системы и компьютерный мониторинг -2010. Материалы V международной научно-технической конференции студентов, аспирантов и молодых ученых.

[2] Аноприенко А.Я., Плотников Д.Ю. , Мальований С.Ф. Использование клеточных автоматов для моделирования вероятностных ситуаций. Информатика и компьютерные технологии -2010. Материалы VI

международной научно-технической конференции студентов, аспирантов и молодых ученых.