

МОДЕЛИРОВАНИЕ И ВИЗУАЛИЗАЦИЯ СИСТЕМЫ СВЕТОФОРНОГО РЕГУЛИРОВАНИЯ НА ПЕРЕКРЁСТКЕ

Кораблев Р.А.¹, Белокуров В.П.¹, Бусарин Э.Н.¹,
Стасюк В.В.¹, Гасилова О.С.², Кораблев А.Р.¹

¹ФГБОУ ВО «Воронежский государственный
лесотехнический университет имени Г.Ф. Морозова»,
г. Воронеж, Россия

²ФГБОУ ВО «Уральский государственный
лесотехнический университет»,
г. Екатеринбург, Россия

Аннотация. С ростом урбанизации возникают транспортные заторы, а также увеличивается интенсивность движения на перекрёстках, что само по себе становится отдельной проблемой, требующей решения. Одним из важнейших факторов при оптимизации цикла светофорного регулирования и определении плотности транспортных потоков, ожидающих на перекрёстках, является задержка. Традиционные методы определения задержки неэффективны при высокой интенсивности движения на перекрёстках. Поэтому объём трафика, время ожидания и среднее количество автомобилей в очереди также являются важными параметрами при проектировании систем светофорного регулирования.

Ключевые слова: светофорное регулирование, перекресток, модель, программа, интерфейс.

MODELING AND VISUALIZATION OF THE TRAFFIC LIGHT CONTROL SYSTEM AT THE INTERSECTION

Korablev R.A.¹, Belokourov V.P.¹, Busarin E.N.¹,
Stasyuk V.V.¹, Gasilova O.S.², Korablev A.R.¹

¹Voronezh State University of Forestry and Technologies
named after G.F. Morozov,
Voronezh, Russia

²Ural State Forestry University,
Ekaterinburg, Russia

Abstract. With increasing urbanization, traffic congestion occurs, as well as increased traffic at intersections, which in itself becomes a separate problem that needs

to be addressed. One of the most important factors in optimizing the traffic light regulation cycle and determining the density of traffic flows waiting at intersections is delay. Traditional delay detection methods are ineffective when there is high traffic at intersections. Therefore, traffic volume, waiting time, and the average number of cars in line are also important parameters in the design of traffic light control systems.

Keywords: traffic light regulation, intersection, model, program, interface.

Исследование направлено на детальный анализ ключевых параметров, определяющих эффективность работы светофорного регулирования на перекрёстках. В частности, будет проведена оценка продолжительности работы светофорных циклов, времени ожидания зелёного сигнала, а также среднего количества транспортных средств в очереди. Результаты исследования позволят разработать и оптимизировать системы управления дорожным движением, что, в свою очередь, будет способствовать более эффективному решению проблем, связанных с заторами и интенсивностью движения в городских районах.

Системы управления дорожным движением играют ключевую роль в обеспечении безопасности участников дорожного движения и повышении пропускной способности на перекрёстках. Однако неправильно настроенные или неэффективные системы могут привести к сбоям и авариям, образованию длинных очередей, длительным задержкам и нарушению правил дорожного движения. Для предотвращения негативных последствий, таких как заторы, аварии и нарушения правил дорожного движения, необходимо правильно настроить и оптимально применять системы управления дорожным движением. В системах управления дорожным движением чаще всего используются два метода: фиксированная синхронизация и адаптивное управление сигналами. В системе фиксированной синхронизации время работы светофоров и последовательность фаз заранее определяются, и система работает в этом режиме постоянно. В системе адаптивного управления сигналами время работы светофоров и последовательность фаз могут динамически регулироваться в зависимости от интенсивности движения в разные часы суток. Среди этих двух систем наиболее часто предпочтение отдается адаптивным системам управления

сигналами. В настоящее время ведутся активные исследования по разработке и совершенствованию адаптивных систем управления сигналами. В исследовании была создана модель и симуляция системы управления дорожным движением для перекрёстков, которая может помочь в проектировании систем сигнализации. С помощью этой модели, предназначенной для четырёх полосных перекрёстков, можно оценить производительность системы светофорного регулирования, сравнивая плотность движения, задержки и время ожидания.

В области управления транспортными потоками и системами сигнализации было проведено множество исследований, направленных на оптимизацию работы дорожных знаков и светофоров. В этих исследованиях широко применялись искусственные нейронные сети и системы управления, основанные на принципах нечеткой логики, для создания эффективных алгоритмов принятия решений. В 1977 году Паппис и Мандани предложили модель принятия решений для управления транспортными потоками, учитывая такие параметры, как длина очереди, порядок приближения и время продления [1]. В 1995 году Тес и его коллеги разработали систему управления дорожными сигналами для сложных транспортных сетей [2]. Тес, МакШейн и Ким предложили модель, основанную на принципах нечеткой логики, и провели сравнительный анализ с системой постоянных дорожных сигналов. Чонван Ким разработал модель сравнения для отдельных и четырехполосных перекрестков с использованием разработанного им алгоритма управления, чтобы определить различия между этими перекрестками и перекрестками с фиксированной синхронизацией сигналов [3]. Дж. П. Нииттымаки разработал алгоритм управления для отдельных и двух полосных перекрестков и сравнил его с адаптивным управлением движением по критериям производительности, таким как среднее время задержки и процент остановок. В результате были внесены улучшения в эти алгоритмы [4]. Хойер и Джумар разработали модель, учитывающую переменный режим фаз, и провели сравнения по настройке

временных интервалов фаз в зависимости от объема трафика в полосах движения и продолжительности красного сигнала светофора [5].

В данной статье была разработана модель управления светофорами для четырехполосного перекрестка с использованием языка программирования C++. Для создания модели применялись алгоритмы конечных автоматов (Finite State Machines) [6-9].

Конечный автомат (Finite State Machine, FSM) - это модель, состоящая из определённого или ограниченного числа состояний, переходов между ними и выполняемых действий. Он сохраняет информацию о предыдущем состоянии, показывая изменения входных данных от начала до текущего момента. Конечный автомат демонстрирует переходные состояния и условия, необходимые для обеспечения этого перехода.

Действие - это активность, выполняемая в определённый момент времени. Существует несколько типов действий:

- Выходное действие - действие, выполняемое при выходе из состояния.
- Входное действие - действие, выполняемое в зависимости от текущего состояния и входных условий.
- Переходное действие - действие, возникающее при выполнении определённого перехода.

Конечные автоматы представляются в виде диаграмм состояний (или диаграмм переходов) (табл. 1). Кроме того, используется множество различных типов таблиц переходов состояний. Наиболее распространенное представление показано ниже: если условие (Y) выполняется в текущем состоянии (B), то следующим состоянием будет (C). Вся информация о действиях может быть добавлена только с помощью примечаний. Определение конечного автомата, включающее всю информацию о действиях, возможно с использованием диаграмм состояний [6,7,8,9].

Таблица 1 - Таблица перехода состояний

Условие	Текущее состояние		
	Состояние А	Состояние В	Состояние С
Условие Х	...	...	...
Условие Y	...	Состояние С	...
Условие Z	...	...	...

Помимо моделирования реактивных систем, конечные автоматы играют значимую роль в различных областях, таких как электротехника, лингвистика, компьютерные науки, философия, биология, математика и логика. Конечные автоматы представляют собой класс автоматов, функционирующих в теории автоматов и теории вычислений. В компьютерных науках конечные автоматы широко применяются для моделирования поведения прикладных систем, проектирования цифровых систем, разработки программного обеспечения, сетевых протоколов, вычислений и обучения языкам [6, 7, 8, 9].

В теории автоматов и теории вычислений существуют две основные категории автоматов: приёмники (acceptor) и распознаватели (recognizer), а также преобразователи (transducer).

Приёмники (acceptor) предназначены для определения, принадлежит ли входная строка к заданному языку. Они принимают или отвергают входные данные, не изменяя их.

Распознаватели (recognizer) выполняют ту же задачу, что и приёмники, но могут также генерировать выходные данные в зависимости от входной строки.

Преобразователи (transducer) выполняют более сложные операции. Они принимают входную строку, изменяют её и генерируют выходную строку. Преобразователи могут быть как детерминированными, так и недетерминированными.

Автоматы и распознаватели, функционируя в двоичной системе, предоставляют ответ «да» или «нет», определяя, соответствует ли входная

строка заданным критериям. Конечные автоматы должны охватывать все возможные состояния. Если текущее состояние соответствует входной строке, она принимается, в противном случае - отклоняется. Правила, используемые в этом процессе, оперируют символами в качестве входных данных и не предполагают выполнения каких-либо действий.

Машину можно рассматривать как язык, состоящий из строк, которые принимаются или отклоняются ею. В соответствии с определением, языки, принимаемые конечными автоматами, относятся к категории регулярных.

Начальное состояние – это состояние, которое показывает начальную точку или состояние, из которого машина может начать работу. Оно обозначается символом "Start" или может быть обозначено любой другой последовательностью символов.

Принятое состояние - состояние, в котором машина успешно завершила выполнение задачи. Оно представлено двумя символами, которые обозначают начало и конец выполнения. В качестве примера можно рассмотреть детерминированный конечный автомат, который представлен двоичными строками с двумя нулями (рис. 1). Мы видим, что при поступлении нулевого символа слева, автомат переходит в начальное состояние S1. Затем, благодаря внутренней структуре автомата, если он находится в состоянии S1, то при поступлении еще одного нулевого символа он автоматически переходит в состояние S1 снова. Таким образом, гарантируется, что каждая принимаемая строка будет содержать четное количество нулей, и автомат всегда будет принимать строки, состоящие из четного числа нулей [6,7,8,9].

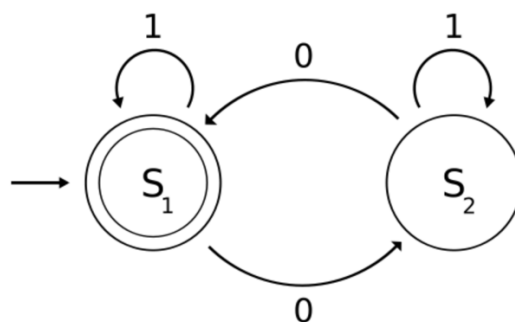


Рисунок 1 – Работа конечного автомата с состояниями

В статье рассматривается модель Х-образного перекрёстка, в которой переменные параметры включают в себя названия улиц, интенсивность движения на них и время ожидания сигналов светофоров. После запуска модели при неожиданной остановке программа предоставляет цифровые и графические статистические данные о количестве транспортных средств, ожидающих на светофорах, а также о среднем и мгновенном времени их ожидания.

Рассмотрим детали программы по разделам.

1. Интенсивность движения и плотность потока

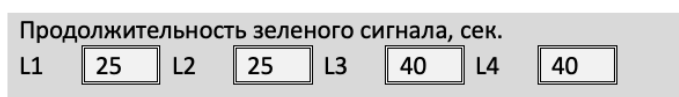
На рис. 2 представлен интерфейс для ввода данных об интенсивности движения на перекрестке. Интенсивность движения на улицах представлена количеством автомобилей, въезжающих на улицу в течение одной минуты. Въезд автомобилей на улицы осуществляется случайным образом в зависимости от количества автомобилей. Динамическое использование названий улиц и интенсивности движения позволяет использовать программу на любом Х-образном перекрёстке и в условиях изменяющейся интенсивности движения.

Улицы и интенсивность дорожного движения, авт./мин.					
		A	20		
B	25			D	40
		C	35		Создать

Рисунок 2 – Интерфейс для ввода данных об интенсивности движения на перекрестке

2. Продолжительность разрешающего сигнала светофора

На рис. 3 представлен интерфейс для ввода данных о продолжительности разрешающего сигнала светофора на перекрестке. Продолжительность разрешающего сигнала светофора на перекрестке представлена временем в секундах, в течение которого горит зеленый сигнал. Включение зеленого сигнала светофора происходит случайным образом в зависимости от продолжительности цикла. Динамическое использование названий улиц и продолжительности сигналов светофора позволяет использовать программу на любом X-образном перекрестке и в условиях изменяющейся продолжительности цикла регулирования.



Продолжительность зеленого сигнала, сек.							
L1	25	L2	25	L3	40	L4	40

Рисунок 3 – Интерфейс для ввода данных о продолжительности разрешающего сигнала светофора на перекрестке

На рис. 3 представлен экран для ввода данных о продолжительности зелёного сигнала светофора на перекрестках. На этом экране можно динамически изменять время ожидания зелёного сигнала для каждого перекрестка в зависимости от интенсивности движения. Таким образом, можно определить оптимальное время ожидания зелёного сигнала для каждого перекрестка. Параметры L1, L2, L3 и L4 обозначают время ожидания зелёного сигнала соответственно для каждого направления.

3. Экран ввода данных о текущей интенсивности движения на перекрестках:

Эти данные позволяют оценить текущую ситуацию, то есть количество автомобилей, проехавших через перекресток на разрешающий сигнал светофора с момента запуска модели.

Интерфейсы ожидания и времени ожидания содержат информацию о количестве транспортных средств, ожидающих на светофорах во время работы модели. Интерфейс ожидания отображает количество транспортных средств,

ожидающих на светофоре с красным светом. Когда сигнал становится зеленым, количество ожидающих транспортных средств уменьшается, и каждое транспортное средство, пересекающее светофор, регистрируется в данных интерфейса.

Интерфейс времени ожидания показывает общее время ожидания транспортных средств на светофоре с красным светом в момент работы модели. Данные на этих интерфейсах обновляются каждый цикл, что позволяет оценивать длину очереди транспортных средств.

Интерфейс состояния светофора предоставляет информацию о состоянии светофоров на улицах в момент работы модели.

4. Интенсивность использования перекрестка

На рис. 4 представлены количественные данные о транспортных средствах, пересекающих перекрёсток на разрешающий сигнал светофора. Например, на улице А 6 транспортных средств направляются на улицу В, 7 транспортных средств - на улицу С и 9 транспортных средств - на улицу D. Направление транспортных средств на определённые улицы может быть случайным или определяться вручную.

	A	B	C	D
A		6	7	9
B	9		12	12
C	15	16		13
D	13	11	13	

Рисунок 4 – Интенсивность использования направлений перекрестка

5. Среднее время ожидания

На экране представлен график, демонстрирующий среднее время ожидания автомобилей у светофоров в ожидании проезда при красном сигнале. Каждый цвет на графике соответствует определённой улице.

6. Средние значения

На рис. 5 представлены данные о среднем количестве транспортных средств, ожидающих на светофоре с запрещающим сигналом, и среднем времени ожидания транспортных средств на каждой из улиц в течение периода функционирования модели. Эти средние значения позволяют оценить степень соответствия модели условиям перекрёстка в зависимости от переменных, введённых в течение периода функционирования.

Средние значения				
	A	B	C	D
Очередь автомобилей	4	12	10	7
Время ожидания, сек	23	22	25	21

Рисунок 5 – Средние значения модели

В результате моделирования было установлено, что размер очереди, формирующейся при запрещающем сигнале светофора, находится в прямой зависимости от интенсивности дорожного движения на различных участках дороги.

Динамическая корректировка времени ожидания в красном сигнале позволила эффективно отслеживать интенсивность движения. Определение оптимальных временных интервалов ожидания и задержек показало, что эти параметры варьируются в зависимости от интенсивности транспортного потока на конкретных участках дороги.

Поскольку оптимальные временные интервалы ожидания и задержек различаются в зависимости от интенсивности транспортного потока на каждом участке дороги, модель демонстрирует различные характеристики в зависимости от участка, на котором она применяется.

В рамках данного исследования были определены оптимальные временные интервалы ожидания и задержек для Х-образных перекрёстков на основе анализа функционирования системы светофорного регулирования.

Внедрение в существующую систему регулирования движения методов нечёткой логики и искусственных нейронных сетей с использованием входных параметров модели позволит создать систему, способную прогнозировать и принимать решения на основе анализа данных и условий дорожного движения. Это, в свою очередь, позволит более точно определять оптимальные параметры и принимать решения для управления движением.

Список литературы

1. Pappis C.P. A Fuzzy logic controller for a traffic junction / C.P. Pappis, E.H. Mamdani // IEEE Transactions on Systems, Man, and Cybernetics. – 1977. – Vol. 7. – 10. – pp. 707-717. DOI: 10.1109/TSMC.1977.4309605.
2. Tzes A. Expert fuzzy logic traffic signal control for transportation networks / A. Tzes, W.R. McShane, S. Kim // Institute of Transportation Engineers 65th Annual Meeting, Denver USA. 1995. – pp. 154-158.
3. Jongwan K. A Fuzzy logic control simulator for adaptive traffic management / K. Jongwan // Proc IEEE International Conference on Fuzzy Systems. – 1997. – pp. 1519-1524. DOI: 10.1109/FUZZY.1997.619768.
4. Niittymäki J., Installation and experiences of field testing a fuzzy signal controller / J. Niittymäki // European Journal of Operational Research. – 2001. – Vol. 131. – 2. – pp. 273-281. – DOI: 10.1016/S0377-2217(00)00127-2.
5. Hoyer R. Fuzzy control traffic control of traffic lights / R. Hoyer, U. Jumar // Proc. IEEE International Conference on Fuzzy Systems. – 1994. – pp. 1526-1531. – DOI: 10.1109/FUZZY.1994.343921.
6. Бобырь М.В. Система управления интеллектуальным светофором на основе нечеткой логики / М.В. Бобырь, Н.И. Храпова, М.А. Ламонов // Известия Юго-Западного государственного университета. – 2021. – 25 (4). – pp. 162-176. – DOI: 10.21869/2223-1560-2021-25-4-162-176.
7. Modeling Software with Finite State Machines: A Practical Approach / F. Wagner, R. Schmuki, T. Wagner, P. Wolstenholme // Auerbach Publications. – 2006. – 392 p. – DOI: 10.1201/9781420013641.
8. Samek M. Practical Statecharts in C/C++: Quantum Programming for Embedded Systems / M. Samek // San Francisco, New York, Lawrence, KSCMP Books. – 2002. – 387 p. - DOI: 10.1201/9781482280807.
9. Cassandras C. Introduction to Discrete Event Systems / C. Cassandras, S. Lafortune // Kluwer, Springer New York. – 2010. – 772 p. – DOI: 10.1007/978-0-387-68612-7.

References

1. Pappis C.P. A Fuzzy logic controller for a traffic junction / C.P. Pappis, E.H. Mamdani // IEEE Transactions on Systems, Man, and Cybernetics. – 1977. – Vol. 7. – 10. – pp. 707-717. DOI: 10.1109/TSMC.1977.4309605.
2. Tzes A. Expert fuzzy logic traffic signal control for transportation networks / A. Tzes, W.R. McShane, S. Kim // Institute of Transportation Engineers 65th Annual Meeting, Denver USA. 1995. – pp. 154-158.
3. Jongwan K. A Fuzzy logic control simulator for adaptive traffic management / K. Jongwan // Proc IEEE International Conference on Fuzzy Systems. – 1997. – pp. 1519-1524. DOI: 10.1109/FUZZY.1997.619768.
4. Niittymäki J., Installation and experiences of field testing a fuzzy signal controller/ J. Niittymäki // European Journal of Operational Research. – 2001. – Vol. 131. – 2. – pp. 273-281. – DOI: 10.1016/S0377-2217(00)00127-2.
5. Hoyer R. Fuzzy control traffic control of traffic lights / R. Hoyer, U. Jumar // Proc. IEEE International Conference on Fuzzy Systems. – 1994. – pp. 1526-1531. – DOI: 10.1109/FUZZY.1994.343921.
6. Bobyr M.V. Smart traffic light control system based on fuzzy logic / M.V. Bobyr, N.I. Khrapova, M.A. Lamonov // Proceedings of the Southwest State University. – 2021. – 25(4). – pp. 162-176. – DOI: 10.21869/2223-1560-2021-25-4-162-176.
7. Modeling Software with Finite State Machines: A Practical Approach / F. Wagner, R. Schmuki, T. Wagner, P. Wolstenholme // Auerbach Publications. – 2006. – 392 p. – DOI: 10.1201/9781420013641.
8. Samek M. Practical Statecharts in C/C++: Quantum Programming for Embedded Systems / M. Samek // San Francisco, New York, Lawrence, KSCMP Books. – 2002. – 387 p. – DOI: 10.1201/9781482280807.
9. Cassandras C. Introduction to Discrete Event Systems / C. Cassandras, S. Lafortune // Kluwer, Springer New York. – 2010. – 772 p. – DOI: 10.1007/978-0-387-68612-7.