

П. К. Ніколюк¹, В. Ф. Комаров¹, П. П. Ніколюк²
Донецький національний університет імені Василя Стуса¹
Вінницький національний технічний університет²
ІНТЕЛЕКТУАЛЬНЕ ПЕРЕХРЕСТЯ

Базовим елементом у технології регулювання міського трафіку являється перехрестя. Саме цей об'єкт, де перетинаються міські дороги, є і основною причиною а також і джерелом заторів. Тому першочергово необхідно здійснити інтелектуальну регуляцію проїзду транспортних засобів (ТЗ) через окреме перехрестя. Таке регулювання здійснюється з допомогою комп'ютерної програми, що враховує завантаженість напрямків на перехресті та здійснює відповідну корекцію протяжності фаз горіння червоного та зеленого світла.

Ключові слова: розумне перехрестя, п'єзокристалічний датчик, зв'язані ТЗ, цикл роботи світлофора.

П. К. Николюк, В. Ф. Комаров, П. П. Николюк
ИНТЕЛЛЕКТУАЛЬНЫЙ ПЕРЕХРЕСТОК

Базовым элементом в технологии регулирования городского трафика является перекресток. Именно этот объект, где пересекаются городские дороги, есть и основной причиной а также и источником пробок. Поэтому в первую очередь необходимо осуществить интеллектуальную регуляцию проезда транспортных средств (ТС) через отдельный перекресток. Такое регулирование осуществляется с помощью компьютерной программы, которая учитывает загруженность направлений на перекрестке и осуществляет соответствующую коррекцию протяженности фаз горения красного и зеленого света.

Ключевые слова: разумный перекресток, пьезокристаллический датчик, связанные ТС, цикл работы светофора.

P.K. Nikolyyuk, V. Komarov, P.P. Nikolyyuk
INTELLIGENT CROSSROAD

The technology of such intellectual regulation of traffic in the big city, in which there will be no traffic jams, is presented. It is proposed to solve the problem of traffic by controlling the regulation of traffic flows at a separate intersection and within the entire metropolis. The choice of most optimal route is based on a monitoring the situation with traffic in the mode of constant control and correction of the route of each vehicle, which orders the route to the central office of traffic management (COTM). The basic element in the technology of urban traffic regulation is an intersection. It is object where the city roads intersect. The object is the main reason as well as a source of traffic jams. Therefore, first of all, it is necessary to carry out an intelligent regulation of passage of vehicles through a separate intersection. Such regulation is carried out by a computer program which takes into account directions congestions at the crossroads and performs a correction length corresponding phases red and green light.

Key words: intelligent crossroad, piezoelectric sensor, connected vehicles, traffic light cycle.

Постановка проблеми. Дане дослідження є продовженням та уточненням робіт [1] та [2], що розглядають проблему трафіку у великому місті. В першу чергу така проблема пов'язана із заторами на перехрестях і, як наслідок, складністю проїзду кожного ТЗ по вибраному маршруту. З метою запобігання такого явища автори пропонують застосувати мережу «розумних» світлофорів; причому датчики сусідніх світлофорів взаємодіють між собою. Така взаємодія фактично означає, що не тільки світлофори знаходяться під контролем центрального пункту керування трафіком (ЦПКТ), а і ділянки дороги між сусідніми перехрестями. Таким чином з'являється можливість контролювати маршрути всіх ТЗ на дорогах міста. Детально ситуація щодо прокладання маршрутів для кожного ТЗ описана в роботі [2]. Виходячи із всього вищесказаного головна проблема, що вирішуватиметься в даній роботі полягає у поєднанні алгоритмів регулювання руху ТЗ як через окреме перехрестя, так і по всьому місту. Головний акцент приділяється алгоритму регулювання трафіку через окреме перехрестя.

Аналіз останніх досліджень і публікацій. Базовою проблемою регулювання дорожнього руху у мегаполісі є проблема оптимізації проїзду ТЗ через кожне окреме перехрестя. Саме на перехрестях виникають затори. Організувавши ефективний рух через такий об'єкт, досягнемо високої ефективності трафіку по всьому місту. Для організації проїзду ТЗ необхідно їх реєструвати. Причому способів реєстрації є багато [1]. Для вирішення нашої проблеми найбільш ефективним способом такої реєстрації є спосіб на основі п'єзокристалічних датчиків, вмонтованих в полотно дороги.

Компанія International Road Dynamics Inc. розробила досить ефективну конструкцію п'єзоелектричного датчика дорожнього руху RoadTrax BL [3,4]. Чутливі елементи апарату легко монтуються в дорожнє полотно. Чутливість пристрою досить висока – не менше 250 мВ. При

наїзді колісної пари на зону розташування датчика вихідний сигнал вказаної величини достатній для реєстрації. Існує понад 10 видів та способів реєстрації рухомих ТЗ. По фізичному принципу дії типу датчики дорожнього руху поділяються на такі види [5]:

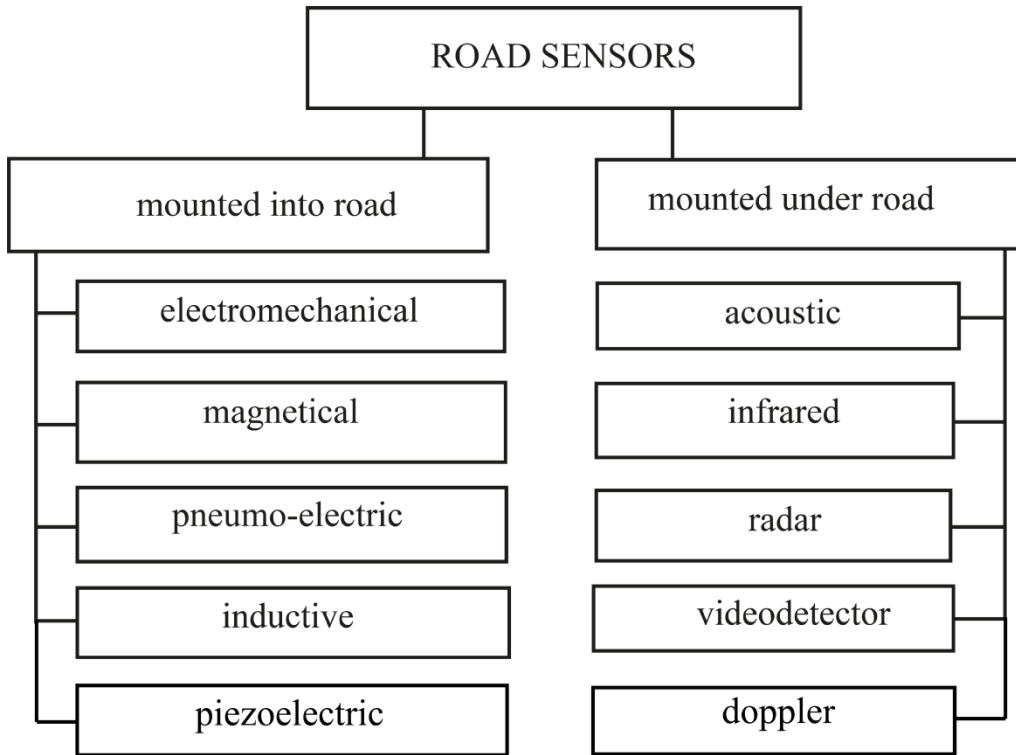


Рис. 1. Класифікація датчиків дорожнього руху згідно дослідження [5].

Найбільш придатними для вирішення поставленої в нашій роботі задачі є датчики, що монтується в дорожнє полотно, а серед цієї групи п'єзокристалічні та індуктивні. Зауважимо, що п'єзоелектричні датчики є більш чутливі, чим індуктивні. Саме тому автори зупинились на високочутливих і ефективних п'єзокристалічних датчиках. В дослідженні [5] приведені технічні характеристики п'єзокристалічного датчика іншої конструкції типу 12-DOF. Це високоякісний високочутливий датчик, реєструючий величини тисків в широкому діапазоні величин. Пристрій працює в частотному діапазоні в околі частоти 1кГц з похибкою не більше 1%.

Теоретичні аспекти оптимізації трафіку на окремому перевантаженому ТЗ перехресті досліджені в роботі [6,7]. Тут розглядається так звана дискретно-часова модель (discrete-time model) для хрестоподібного перехрестя, що записується у вигляді системи рівнянь наступного виду:

$$J_D = \sum_{k=0}^N [q_1(k) + q_2(k)] + \frac{a_1 + a_2}{2} \cdot T \cdot \sum_{k=0}^{N-1} u(k) \rightarrow \min \quad (1)$$

$$q_1(k+1) \geq \max\{q_1(k) + d_1 \cdot T \cdot (u(k) - u_L), a_1 \cdot T(1 - u(k))\} \quad (2)$$

$$q_2(k+1) \geq \max\{q_2(k) + d_2 \cdot T(u_H - u(k)), 0\} \quad (3)$$

$$u_{\min} \leq u(k) \leq u_{\max} \quad (4)$$

$$q_i(0) = q_{i,\text{int}}, i = 1, 2; N = 0, 1, \dots, N-1. \quad (5)$$

В приведених рівняннях фігурують такі змінні:

J_D – модельна величина, що являє собою число ТЗ на перевантаженому перехресті, що не перетнули його (total delay);

N – число циклів перемикання світлофора;

k – нумератор циклів;

$q_1(k), q_2(k)$ – число ТЗ на перехресті в k -му циклі переключення світлофора відповідно на горизонтальному та вертикальному напрямках хрестоподібного перехрестя;

a_1, a_2 – число ТЗ, що прибули за одиницю часу відповідно на горизонтальний та вертикальний напрямки;

T період перемикання світлофору (цикл роботи);

$u(k)$ – відношення величини фази горіння зеленого світла до аналогічної величини для червоної фази;

d_1, d_2 – число ТЗ, що перетнули перехрестя за одиницю часу;

$u_H = 1 - a_2/d_2, u_L = a_1/d_1$;

u_{\min}, u_{\max} – відповідно мінімальне та максимальне значення $u(k)$;

$q_{i,\text{int}}$ – початкова величина ТЗ на перехресті для горизонтального ($i = 1$) та вертикального напрямків ($i = 2$).

Цільова функція системи рівнянь (1)-(5) виражається виразом (1) і являє собою число ТЗ, що не перетнули перехрестя. Зрозуміло, що таку величину треба мінімізувати. Шуканими змінними є величини $u(k)$. Рівняння (2) і (3) представляють еволюцію з часом обох автомобільних черг. Рівняння (4) дає верхню і нижню границі $u(k)$ в кожному циклі. Рівняння (5) представляє стартові (початкові) довжини автомобільних черг на перехресті. Загалом система рівнянь (1)-(5) являє собою задачу лінійного програмування. Графічний розв'язок цієї задачі представлений на рис.2.

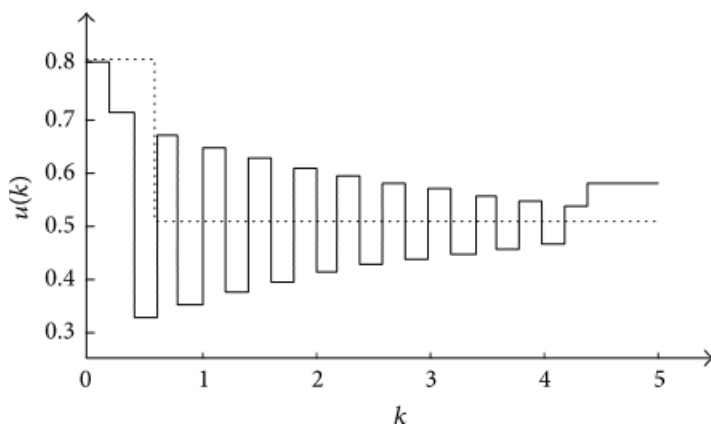


Рис. 2. Розв'язок системи рівнянь (1)–(5): суцільна лінія – точний розв'язок; точкова – наближений [7].

Дослідження методів синтезованої інформації, отримуваної із різноманітних джерел, таких як датчики, детектори, відеокамери і радари проведено в роботі [9]. Тут розглядається технологія зв'язаних ТЗ (connected vehicles), яка дає можливість збирати та аналізувати зв'язки типу «ТЗ – Інфраструктура» (V2I) та «ТЗ – ТЗ» (V2V). Зв'язки такого типу дають можливість зменшити імовірність заторів, збільшити безпеку руху та зменшити витрати палива.

На рис.2 зображена схема взаємодії автомобілів, що проїжджають перехрестя, із придорожною інфраструктурою. Система передачі даних OBU забезпечує V2X комунікацію, тобто взаємодію автомобілів з придорожною інфраструктурою.

Постановка завдань. З аналітичної сторони необхідно створити алгоритм та комп'ютерну програму, що забезпечують ефективне перемикання світлофорних фаз у відповідності із завантаженістю дорожніх напрямків. З технічного боку реєстрація ТЗ здійснюється з допомогою п'єзокристаличних датчиків, які фіксують число колісних пар, пропорційне числу ТЗ, що або

в'їжджають на проїзну частину дороги між сусідніми перехрестями (так звані входні датчики) або виїжджають із цієї ділянки дороги – вихідні датчики.

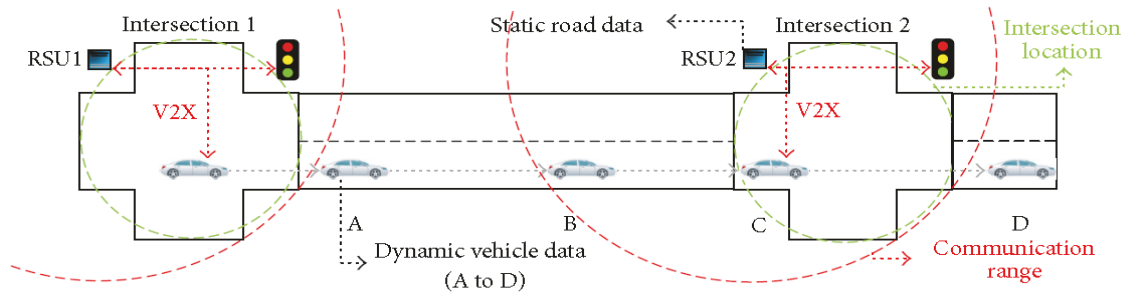


Рис.3. Технологія зв'язаних автомобілів (CV), де RSU – Road Side Unit, тобто придорожня комірка, яка взаємодіє як із світлофором, так і з автомобілем. Діапазон взаємодії CV із RSU зображений у вигляді колової траєкторії [8]. На автомобілях встановлена система передачі даних OBU (On Board Unit).

При цьому входні та вихідні датчики на сусідніх перехрестях працюють сумісно. Це дозволяє контролювати не тільки міські перехрестя як автономні об'єкти, а також і всі міські траси.

Викладення основного матеріалу

Реєстрація ТЗ, перетинаючих перехрестя, організована наступним чином (рис.4).

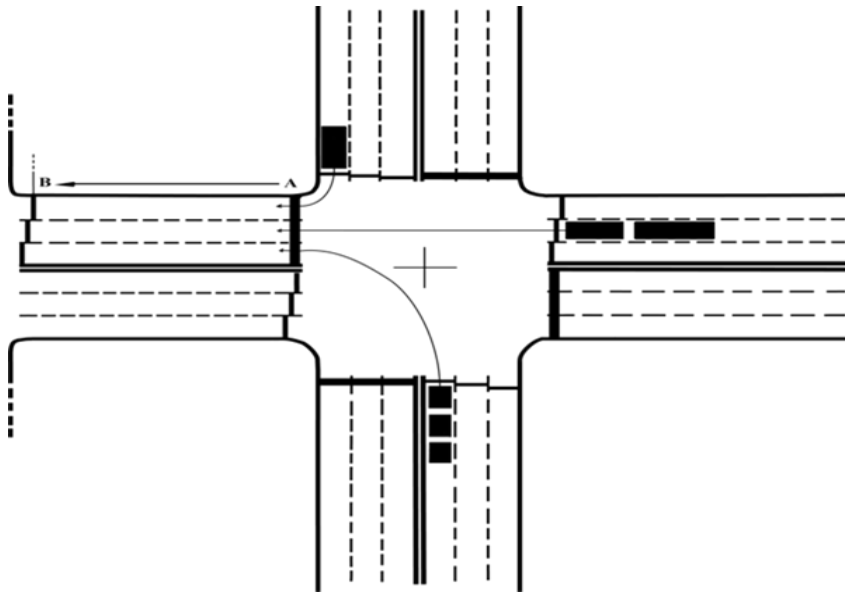


Рис.4. Перехрестя А, з'єднане з сусіднім перехрестям В [1]. Чорними прямокутниками зображені ТЗ, маршрути яких пролягають в напрямку $A \rightarrow B$. Показані також входні (чорна смужка, позначена літерою А) та вихідні датчики (три рядом розташовані смужки, позначені літерою В). Кожен входний та вихідний датчики з'єднані з ЦПКТ (на рисунку зображений лише один такий сигнальний провід, що виходить від одного із датчиків В – це суцільна лінія, що переходить в пунктирну).

Зауважимо, що зображена на рис.4 зліва ділянка дороги $A \rightarrow B$ між сусідніми перехрестями дозволяє контролювати всю транспортну мережу міста, адже тут представлена ділянка дороги між сусідніми перехрестями, що знаходиться під контролем входних та вихідних датчиків. Іншими словами, кожне перехрестя працює не автономно, а у тісній взаємодії із сусідніми перехрестями. Така взаємодія здійснюється за рахунок того, що входні та вихідні датчики (рис. 4) працюють в парі – як один елемент. За рахунок цього можливо прослідкувати за динамікою руху ТЗ вздовж ділянки дороги $A \rightarrow B$. На якісному рівні динаміка руху в даному разі обумовлена співвідношенням між величиною ТЗ, що в'їхали за певний час (наприклад, за час,

рівний циклу переключення світлофора) та виїхали із даної ділянки дороги. Технічно процес організований наступним чином: датчик А (вхідний датчик) реєструє автомобілі, які в'їжджають на ділянку дороги АВ з усіх можливих напрямків перехрестя А. В свою чергу комплекс вихідних датчиків на перехресті В реєструє автомобілі, що виїхали за межі цієї ділянки.

Фактично наявність вхідних та вихідних датчиків дозволяє сформувавши запис виду

$$\sum_{h=1}^f (N_{A_h B_h} / n_{A_h B_h}) l_{A_h B_h} \rightarrow \min. \quad (6)$$

Тут h – індекс, що позначає смуги руху вздовж маршруту, що зчеплені між собою та утворюють нерозривну траєкторію. Символ f означає число смуг виду $A_h B_h$, які формують прокладений маршрут. $N_{A_h B_h}$ – число автомобілів, що в'їжджають на ділянку дороги $A \rightarrow B$ між сусідніми перехрестями, а $n_{A_h B_h}$ – відповідно число автомобілів, що виїжджають із цієї ділянки дороги та реєструються тріадою вихідних датчиків (рис.4). Величина $l_{A_h B_h}$ являє собою просто геометричну довжину дороги між сусідніми перехрестями.

Отже, в основі всієї технології регулювання трафіку як вихідна комірка слугує окреме регульоване перехрестя. Вважатимемо, що цикл роботи цього світлофора – це величина, що, як правило, складається з таких складових

$$T = trh + tgh + tyh + tp, \quad (7)$$

де trh – час червоного світла у горизонтальному напрямку;

tgh – час горіння зеленого світла у горизонтальному напрямку;

tyh – час горіння жовтого світла також у горизонтальному напрямку;

tp – час горіння зеленого світла для пішоходів.

Таке саме співвідношення можна записати по відношенню до вертикального напрямку перехрестя, тобто

$$T = trv + tgv + tyv + tp, \quad (8)$$

де trv – час червоного світла у вертикальному напрямку;

tgv – час горіння зеленого світла у вертикальному напрямку;

tyv – час горіння жовтого світла так само у вертикальному напрямку;

tp – час горіння зеленого світла для пішоходів.

Зауважимо, що важливим чинником покращення надійності та достовірності даних, отримуваних на основі аналізу результатів вхідних та вихідних датчиків, є інформація, передана GPS-пристроями. Справа в тому, що майже кожен водій працює якщо не з GPS-навігатором, то із мобільним телефоном чи смартфоном з програмою Google-карт. Це дозволяє використати GPS-дані в якості додаткового джерела інформації. Описана ситуація детально розглядається в роботах [10-14]. Важливо зауважити, що GPS-дані не дають точної інформації. Але для нашої задачі такі дані важливі як допоміжний показник щодо інтерпретації даних, отримуваних від дорожніх датчиків. В цьому відношенні з урахуванням специфіки поставленої в роботі задачі, статистичний характер GPS-даних цілком прийнятний і немає необхідності точно визначати положення кожного ТЗ: важливим є лише спектр таких, отримуваних з певною долею достовірності. Чому важливо поєднати дані, отримувані із п'єзокристалічних датчиків, з GPS-даними? Справа в тому, що основну роль в нашій програмі відіграють саме дані, отримувані із датчиків – вхідних та вихідних, – розташованих між сусідніми перехрестями. З іншого боку, програма передбачає використання водіями GPS-навігаторів. Тому цілком логічно використати GPS-інформацію в повній мірі. На доцільність поєднання стаціонарних даних та мобільних, отримуваних із GPS-пристроїв чи смартфонів, вказується, зокрема, і в роботі [15]. Подібний алгоритм використання GPS-даних для прокладання оптимальних маршрутів детально розглядається також в дослідженні [16]. Тут використовуються набори GPS-даних, що характеризуються географічною широтою, довготою та часовим маркером. GPS-точки в даній роботі генеруються з певною вибраною частотою імітатором і на основі них прокладається оптимальний маршрут а також обраховується середня швидкість кожного ТЗ.

Тепер представимо сам алгоритм регулювання трафіку на окремому перехресті, в якому використані змінні, описані під формулами (7) та (8). З використанням приведених змінних програму регуляції трафіку можна записати у наступному вигляді:

```
package IC;
import java.util.Random;
package IC;
import java.util.Random;
import static java.lang.StrictMath.abs;
interface Lights{
int REDH =0;
int YELLOWH =1;
int GREENH =2;
int REDV =3;
int YELLOWV =4;
int GREENV =5;
int GREENP = 6;
int ERROR = -1;
}
class T implements Lights {
private int delay;
private static int light = REDH;
T(int sec) {
delay = 1000 * sec;
}
public int shift() {
int count = (light++) % 7;
try {
switch (count) {
case REDH:
Thread.sleep(delay);
break;
case YELLOWH:
Thread.sleep(delay / 3);
break;
case GREENH:
Thread.sleep(delay / 2);
break;
case REDV:
Thread.sleep(delay);
break;
case YELLOWV:
Thread.sleep(delay / 3);
break;
case GREENV:
Thread.sleep(delay / 2);
break;
case GREENP:
Thread.sleep(delay );
break;
}
} catch (Exception e) {
return ERROR;
}
return count;
} }
class TrafficRegulator {
static int T = 96;
```

```

private static T t = new T(1);
static Random gn1 = new Random();
static Random gn2 = new Random();
static Random gn3 = new Random();
static Random gn4 = new Random();
TrafficRegulator() {
}
public static void main(String[] args) {
double k = Math.abs ((gn1.nextDouble() + gn2.nextDouble() ) /
    (gn3.nextDouble() + gn4.nextDouble()));
double tg = 35.0;
double tp = 23.0;
double tyh = 2.0;
double tgh = k * tg;
double trh = (double)T - tyh - tgh - tp;
double tgv = abs(2.0 * tg - tgh);
double trv = (double)T - trh;
int tyv = 2;
for(int j = 0; j < 7; ++j) {
switch(t.shift()) {
case -1:
System.out.println("Time error!");
break;
case 0:
System.out.println("red horizontal!");
System.out.format("%.1f%n", trh);
break;
case 1:
System.out.println("yellow horizontal!");
System.out.println(tyh);
break;
case 2:
System.out.println("green horizontal!");
System.out.format("%.1f%n", tgh);
break;
case 3:
System.out.println("red vertical!");
System.out.format("%.1f%n", trv);
break;
case 4:
System.out.println("yellow vertical!");
System.out.println(tyv);
break;
case 5:
System.out.println("green vertical!");
System.out.format("%.1f%n", tgv);
break;
case Lights.GREENP:
System.out.println("green pedestrian!");
System.out.println(tgp);
break;
default:
System.err.println("Unknown light.");
return;
}}}}

```

В результаті роботи програми отримуємо:

red horizontal!
29,6
yellow horizontal!
2.0
green horizontal!
41,4
red vertical!
66,4
yellow vertical!
2
green vertical!
28,6
green pedestrian!
23.0

Process finished with exit code 0.

Зупинимось коротко на аналізі даної програми, яка регулює проїзд ТЗ через хрестоподібне перехрестя. В якості констант вибираються величини типу $REDH = 0$ – червоне світло в горизонтальному напрямку. Принципово те, що програма аналізує завантаженість ТЗ напрямків перехрестя: чим більше завантажений напрямок – горизонтальний чи вертикальний – тим більше горітиме зелене світло у відповідному напрямку, звичайно в межах періоду перемикавання світлофора, який в даному разі вибраний рівним 96 с. З метою імітації реальної ситуації на перехресті, в програмі включено чотири генератори випадкових величин, що імітують завантаженість перехрестя ТЗ. Власне, нас цікавить відношення числа автомобілів, що розташовані на горизонтальному та вертикальному напрямках. В програмі таке відношення задається величиною коефіцієнта k . Варіація величини k імітує якраз відношення у змінах завантажених. Тому спектр чисел на виході програми щораз змінюється випадковим чином, моделюючи реальну динаміку трафіку на перехрестях міст. Змінюються, зокрема, величини, що задають час горіння зеленого світла у горизонтальному та вертикальному напрямках.

Отже, презентована програма дає можливість суттєво покращити пропускну здатність перехрестя завдяки «розумному» режиму їх роботи в плані протяжності горіння різних фаз.

Висновки. Цікаве висловлення приведене в роботі [10]. Приведемо його дослівно: «However, to the best of our knowledge, no research has been conducted to apply deep learning theory into large-scale transportation network modeling and analysis.» В перекладі даний вираз звучить так: «Однак, наскільки нам відомо, не проводилось ніякого дослідження з метою застосування теорії в моделюванні та аналізі широкомасштабної транспортної мережі». Ми в певній мірі усунули цей недолік. Робота носить практичний характер, а її результати можуть бути використані на реальних міських перехрестях з метою покращення умов проїзду як через окреме перехрестя, так і через все місто.

Список використаних джерел

1. Богуто Д.Г., Волинець В.І, Ніколюк П.К., Ніколюк П.П. Автоматизована система керування рухом транспортних засобів в межах міста / Д.Г. Богуто, В.І. Волинець, П.К. Ніколюк, П.П. Ніколюк // Вісник Харківського університету, серія «Математичне моделювання. Інформаційні технології. Автоматизовані системи управління». – 2017. – №5. – С. 3-9.
2. Богуто Д.Г., Комаров В.Ф., Ніколюк П.К., Ніколюк П.П. Інтелектуальний алгоритм управління міським трафіком/ Д.Г. Богуто, В.Ф. Комаров, П.К. Ніколюк, П.П. Ніколюк // Вісник Харківського університету, серія «Математичне моделювання. Інформаційні технології. Автоматизовані системи управління». – 2018. – №38. – С. 46-54.
3. Piezoelectric RoadTrax BL - International Road Dynamics, Inc. [Електронний ресурс]. – Режим доступу: www.irdinc.com/sensors/piezoelectric-roadtrax-bl.html.
4. Датчики дорожнього движения – Студопедия. [Електронний ресурс]. – Режим доступу: https://studopedia.ru/7_31480_datchiki-dorozhnogo-dvizheniya.html.
5. Jun Liu, Chuan-Wei Liang, Min Li, Ke Jian, Lan Qin, and Jing-Cheng Liu. Principle Research on a Novel Piezoelectric 12-DOF Force/Acceleration Sensor/Liu Jun , Liang Chuan-Wei, Li Min, Jian Ke, Qin Lan , and Liu Jing-heng//Journal of Sensors.–V.2017. – Article ID 2836365. –16 pages.

6. Yingfeng Cai, Ze Liu, Xiaoqiang Sun, Long Chen, Hai Wang, and Yong Zhang. Vehicle Detection Based on Deep Dual-Vehicle Deformable Part Models/ Cai Yingfeng, Liu Ze, Sun Xiaoqiang, Chen Long, Wang Hai, and Zhang Yong //Journal of Sensors. – Volume 2017. – Article ID 5627281. – 10 pages.
7. Jiyuan Tan, Xiangyun Shi, Zhiheng Li, Kaidi Yang, Na Xie, Haiyang Yu, Li Wang, Zhengxi Li. Continuous and Discrete-Time Optimal Controls for an Isolated Signalized Intersection/ Tan Jiyuan , Shi Xiangyun , Li Zhiheng , Yang Kaidi , Xie Na , Yu Haiyang , Wang Li , Li Zhengxi //Journal of Sensors.– Volume 2017. – Article ID 6290248. – 11 pages.
8. Ilya Ioslovich, Jack Haddad, Per-Olof Gutman, David Mahalel. Optimal traffic control synthesis for an isolated intersection/ Ilya Ioslovich, Jack Haddad, Per-Olof Gutman, David Mahalel// Control Engineering Practice.–2011.–V.19.– Is.8.– P. 900-911.
9. Pang-wei Wang, Hong-bin Yu, Lin Xiao, Li Wang. Online Traffic Condition Evaluation Method for Connected Vehicles Based on Multisource Data Fusion/ Wang Pang-wei, Yu Hong-bin, Xiao Lin, Wang Li // Journal of Sensors. – Volume 2017. – Article ID 7248189. – 11 pages.
10. Mohammed Quddu, Simon Washington. Shortest path and vehicle trajectory aided map-matching for low frequency GPS data/ Quddu Mohammed, Washington Simon // Transportation Research Part C.– 2015.–V.55.–P.328-339.
11. Sina Dabiri, Kevin Heaslip Inferring transportation modes from GPS trajectories using a convolutional neural network/ Dabiri Sina, Heaslip Kevin //Transportation Research Part C. – V. 86.– 2018. – P. 360–37.
12. Feilong Wang, Cynthia Chen On data processing required to derive mobility patterns from passively-generated mobile phone data/ Wang Feilong, Chen Cynthia // Transportation Research Part C. – V. 87.– 2018. – P. 58–74.
13. X. Ma, H. Yu, Y. Wang, Y. Wang, and J. Gomez-Gardenes/Large-scale transportation network congestion evolution prediction using deep learning theory. Ma X., Yu H., Wang Y., Wang Y., Gomez-Gardenes J. //PLoS ONE. – 2015.–V. 10. – №3.– Article ID e0119044.–17 p.
14. Mahdi Hashemi, Hassan A. Karimi. A weight-based map-matching algorithm for vehicle navigation in complex urban networks/ Hashemi Mahdi, Karimi A. Hassan//J. of Intel. Transp. Systems. – 2016. –V.20. –Is.6. – P.45-76.
15. Mahmood Rahmani, Eric Jenelius, Harilaos N. Koutsopoulos. Floating car and camera data fusion for non-parametric route travel time estimation/ Rahmani Mahmood, Jenelius Eric, Koutsopoulos N. Harilaos //Procedia Comp. Science. – 2014. – V.37. –P. 390-395.
16. Masoud Fadaei Oshyaniv, Marcus Sundberg, Anders Karlström. Consistently estimating link speed using sparse GPS data with measured errors/ Masoud Fadaei Oshyaniv, Marcus Sundberg, Anders Karlström// Procedia – Social and Behavioral Sciences. – 2014. – V.111. – P. 829-838.

Рецензенти:

1. д.т.н., професор кафедри радіофізики та кібербезпеки Донецького національного університету імені Василя Стуса Крижановський Володимир Григорович (внутрішній рецензент).
2. д.т.н., професор, декан факультету комп'ютерних систем і автоматики Вінницького національного технічного університету Бісікало Олег Володимирович (зовнішній рецензент).