

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

ННК «Інститут прикладного системного аналізу»
(повна назва інституту/факультету)

Кафедра системного проектування
(повна назва кафедри)

«На правах рукопису»
УДК 004.852

«До захисту допущено»

Завідувач кафедри
_____ А.І. Петренко
(підпис) (ініціали, прізвище)

“ _____ ” _____ 20__ р.

Магістерська дисертація

зі спеціальності (спеціалізації) 8.05010103 Системне проектування
(код і назва спеціальності)

на тему: Оптимізація позиціонування сенсорів в системі Smart Parking

Виконав: студент 6 курсу, групи ДА-61м
(шифр групи)

_____ Куц Михайло Сергійович _____
(прізвище, ім'я, по батькові) (підпис)

Науковий керівник _____ доц., к.т.н., Чкалов О.В. _____
(посада, науковий ступінь, вчене звання, прізвище та ініціали) (підпис)

Консультант _____ Розроблення стартап-проекту _____
(назва розділу) (науковий ступінь, вчене звання, прізвище, ініціали) (підпис)

Рецензент _____
(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище та ініціали) (підпис)

Засвідчую, що у цій магістерській
дисертації немає запозичень з праць
інших авторів без відповідних посилань.

Студент _____
(підпис)

Київ – 2018 року

**Національний технічний університет України
«Київський політехнічний інститут
імені Ігоря Сікорського»**

Інститут/факультет ННК «Інститут прикладного системного аналізу»
(повна назва)

Кафедра _____ Системного проектування _____
(повна назва)

Рівень вищої освіти – другий (магістерський) за освітньо-професійною (освітньо-науковою) програмою

Спеціальність (спеціалізація) 122 Комп'ютерні науки та інформаційні системи
(Інформаційні системи та технології проектування)
(код і назва)

ЗАТВЕРДЖУЮ
Завідувач кафедри

(підпис) А.І. Петренко
(ініціали, прізвище)
« ____ » _____ 20__ р.

**ЗАВДАННЯ
на магістерську дисертацію студенту
Куцу Михайлу Сергійовичу
(прізвище, ім'я, по батькові)**

1. Тема дисертації «Оптимізація позиціонування сенсорів в системі Smart Parking»

науковий керівник дисертації Чкалов О.В., к.т.н., доц.,
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від « ____ » _____ 20__ р. № _____

2. Строк подання студентом дисертації _____

3. Об'єкт дослідження Система Smart Parking

4. Предмет дослідження Алгоритми оптимального позиціонування сенсорів в системі Smart Parking

5. Перелік завдань, які потрібно розробити ознайомитися із загальною проблематикою Smart Parking, провести огляд елементної бази сенсорів, проаналізувати існуючі системи Smart Parking та досвід їх розгортання, проаналізувати існуючі алгоритми та підходи до задачі оптимізації розташування

сенсорів, розробити алгоритм для вирішення задачі оптимального позиціонування сенсорів, оформити роботу на основі отриманих результатів

6. Орієнтовний перелік публікацій Куц М.С. Топологія мережі системи розумної парковки / М.С. Куц. // Міжнародний науковий журнал "Інтернаука". – 2018. – №9., Куц М.С. Оптимальне розміщення датчиків для системи розумної парковки / М.С. Куц // Міжнародний електронний науковий журнал "Наука Онлайн". – 2018. – №5.

7. Консультанти розділів дисертації*

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Реалізація стартап-проекту			

8. Дата видачі завдання 01.02.2018

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Строк виконання етапів магістерської дисертації	Примітка
1	Отримання завдання	01.02.2018	
2	Збір інформації та аналіз літератури	15.02.2018	
3	Ознайомлення із загальною проблематикою Smart Parking	28.02.2018	
4	Аналіз існуючих алгоритмів та підходів до задачі оптимізації розташування сенсорів.	13.03.2018	
5	Розробка та реалізація алгоритму для оптимального позиціонування сенсорів	25.03.2018	
6	Оформлення дипломної роботи	30.04.2018	
7	Отримання допуску до захисту та подача роботи в ДЕК	10.05.2018	

Студент

(підпис)

Куц М.С.
(ініціали, прізвище)

Науковий керівник дисертації

(підпис)

Чкалов О.В.
(ініціали, прізвище)

* Консультантом не може бути зазначено наукового керівника

РЕФЕРАТ НА МАГІСТЕРСЬКУ ДИСЕРТАЦІЮ

виконану на тему: Оптимізація позиціонування сенсорів в системі Smart Parking.

студентом: Куцом Михайлом Сергійовичем

Робота виконана на 91 сторінці, містить 11 ілюстрації, 24 таблиця. При підготовці використовувалась література з 34 джерел.

Актуальність теми

На сучасних автостоянках відсутня єдина система для моніторингу місць для паркування. Основною складністю створення єдиної глобальної системи Smart Parking є відсутність стандартизації в програмних та мережевих засобах при побудові систем IoT. Така система дозволить суттєво знизити витрати часу та пального при пошуку місця для парковки, адже вона надає можливість проводити моніторинг в реальному часі та бронювати потрібне місце на зазначений час.

Розгортання системи Smart Parking може видатися не вигідним для власників парковок чи для державних підприємств, адже для цього необхідні кваліфіковані мережеві інженери та дороге обладнання. Алгоритм для вирішення задачі оптимального розташування сенсорів зможе суттєво автоматизувати цей процес, дозволивши витратити менше часу при проектуванні мережі для інженерів, що може суттєво скоротити витрати на розгортання системи Smart Parking.

Мета та задачі дослідження

Метою роботи є дослідження та розробка алгоритму для вирішення задачі оптимального позиціонування сенсорів в системі Smart Parking. Тестування алгоритму на різних схемах парковок з перешкодами для поширення сигналу.

Рішення поставлених завдань та досягнуті результати

В ході виконання магістерської роботи було досліджено існуючий прототип системи Smart Parking в Алжирі. Було створено програму, в якій було реалізовано алгоритм для оптимального розташування сенсорів на парковках з довільним ландшафтом.

За допомогою створеної програми було проведено тестування на довільних схемах паркувальних майданчиків. Було проведено аналіз рішень, що пропонує програма та створено тестові ситуації з різною щільністю місць для парковки.

Об'єкт досліджень

Система Smart Parking.

Предмет досліджень

Алгоритми оптимального позиціонування сенсорів в системі Smart Parking.

Методи досліджень

Для вирішення проблеми в даній роботі використовуються методи аналізу і синтезу, системного аналізу, порівняння, логічного узагальнення результатів.

Наукова новизна

Наукова новизна роботи полягає у розробці алгоритму та програми на його основі для вирішення задачі оптимізації позиціонування сенсорів на паркувальних майданчиках з довільним ландшафтом.

Практичне значення одержаних результатів

Отримані результати можуть використовуватись у майбутніх дослідженнях за напрямком дослідження алгоритмів оптимізації розташування сенсорів на певній території з урахуванням її перешкод для обмеження розповсюдження сигналів.

Публікації

Куц М.С. Топологія мережі системи розумної парковки / М.С. Куц. // Міжнародний науковий журнал "Інтернаука". – 2018. – №9.

Куц М.С. Оптиміальне розміщення датчиків для системи розумної парковки / М.С. Куц // Міжнародний електронний науковий журнал "Наука Онлайн". – 2018. – №5.

Ключові слова

Smart Parking, інтернет речей, бездротова сенсорна мережа, RFID, комутатор, сенсор.

РЕФЕРАТ НА МАГИСТЕРСКУЮ ДИССЕРТАЦИЮ

исполненную по теме: Оптимизация расположения сенсоров в системе Smart Parking

студентом: Куцом Михаилом Сергеевичем

Работа исполнена на 91 страницах, содержит 11 иллюстрации, 24 таблицы. При подготовке использовалась литература из 34 источников.

Актуальность темы

На современных автостоянках отсутствует единая система для мониторинга мест для парковки. Основной сложностью создания единой глобальной системы Smart Parking является отсутствие стандартизации в программных и сетевых средствах при построении систем IoT. Такая система позволит существенно снизить затраты времени и горючего при поиске места для парковки, ведь она позволяет проводить мониторинг в реальном времени и бронировать нужное место на указанное время.

Развертывание системы Smart Parking может показаться невыгодным для владельцев парковок или для государственных предприятий, ведь для этого необходимы квалифицированные сетевые инженеры и дорогостоящее оборудование. Алгоритм для решения задачи оптимального расположения сенсоров сможет существенно автоматизировать этот процесс, позволив потратить меньше времени при проектировании сети для инженеров, может существенно сократить затраты на развертывание системы Smart Parking.

Цели и задачи исследования

Целью работы является исследование и разработка алгоритма для решения задачи оптимального позиционирования сенсоров в системы Smart Parking. Тестирование алгоритма на различных схемах парковок с препятствиями для распространения сигнала.

Решения поставленных задач и полученные результаты

В ходе выполнения магистерской работы были исследованы существующий прототип системы Smart Parking в Алжире. Была создана программа, в которой было реализовано алгоритм для оптимального расположения сенсоров на парковках с произвольным ландшафтом.

С помощью созданной программы было проведено тестирование на произвольных схемах парковок. Был проведен анализ решений, предлагает программа и создан тестовые ситуации с разной плотностью мест для парковки.

Объект исследования

Система Smart Parking.

Предмет исследования

Алгоритмы оптимального позиционирования сенсоров в системе Smart Parking.

Методы исследований

Для решения проблемы в данной работе используются методы анализа и синтеза, системного анализа, сравнения, логического обобщения результатов.

Научная новизна

Научная новизна работы заключается в разработке алгоритма и программы на его основе для решения задачи оптимизации позиционирования сенсоров на парковочных площадках с произвольным ландшафтом.

Практическое значение полученных результатов

Полученные результаты могут использоваться в будущих исследованиях по направлению исследования алгоритмов оптимизации расположения сенсоров на определенной территории с учетом ее препятствий для ограничения распространения сигналов.

Публикации

Куц М.С. Топология сети системы умной парковки / М.С. Куц. // Международный научный журнал "Интернаука". - 2018. - №9.

Куц М.С. Оптимальное размещение датчиков для системы умной парковки /
М.С. Куц // Международный электронный научный журнал "Наука Онлайн". - 2018.
- №5.

Ключевые слова

Smart Parking, интернет вещей, беспроводная сенсорная сеть, RFID,
коммутатор, датчик.

ABSTRACT ON MASTER'S THESIS

on topic: Optimization of sensors positioning in the Smart Parking system

student: Kuts Mykhailo

Work carried out on 91 pages containing 11 figures, 24 tables. The paper was written with references to 34 different sources.

Topicality

Single system for monitoring parking spaces is missing on the modern car parks. The main difficulty of creating a single global Smart Parking system is the lack of standardization in software and networking tools when building IoT systems. Such a system will significantly reduce the time and fuel costs when searching for a parking space, as it provides real-time monitoring and reservation of the desired location at the specified time.

The deployment of Smart Parking may seem disadvantageous for parking owners or for state-owned enterprises, as it requires qualified network engineers and expensive equipment. The algorithm for solving the problem of optimal sensor location can substantially automate this process by allowing less time to be spent on designing a network for engineers, which can significantly reduce the cost of deploying the Smart Parking system.

Purpose

The purpose of the work is to research and develop an algorithm for solving the problem of optimal positioning of sensors in the Smart Parking system. Testing of the algorithm on different schemes of parking with obstacles for signal propagation.

Solution

During the implementation of the master's thesis, an existing prototype of the Smart Parking system in Algeria was studied. A program was created in which an algorithm for optimal positioning of sensors in parking lots with an arbitrary landscape was implemented.

With the help of the created program, testing was carried out on arbitrary schemes of parking areas. An analysis of the solutions offered by the program was created and test situations were created with different density of parking spaces.

Object of research

Smart Parking System.

Subject of research

Algorithms of sensors positioning optimization in the Smart Parking system

Research methods

To solve the problem in this paper we use methods of analysis and synthesis, system analysis, comparison, logical generalization of the results.

Scientific novelty

The scientific novelty of the work is to develop an algorithm and program on its basis to solve the problem of optimizing the positioning of sensors in parking areas with an arbitrary landscape.

The practical value of the results

The obtained results can be used in future research in the direction of studying algorithms for optimizing the location of sensors in a certain area, taking into account its obstacles to limit the propagation of signals.

Publications

Kuts M.S. Topology of the network of intelligent parking / M.S. Kuts // International scientific magazine "Internet Science". - 2018 - №9.

Kuts M.S. Optimal placement of sensors for intelligent parking system / M.S. Kuts // International scientific journal "Science Online". - 2018 - №5.

Keywords

Smart Parking, Internet Speech, Wireless Sensor Network, RFID, Switch, Sensor.

ЗМІСТ

Перелік умовних позначень	13
ВСТУП	14
1 СИСТЕМИ SMART PARKING	16
1.1 Визначення.....	16
1.2 Історія Smart Parking	17
1.3 Основні виклики Smart Parking.....	18
1.4 Класифікація систем Smart Parking.....	19
1.4.1 Системи на основі агентної моделі (agent model)	19
1.4.2 Системи на основі нечіткої логіки	19
1.4.3 Системи на основі бездротових сенсорних мереж	20
1.4.4 Системи на основі Vehicle to Infrastructure interaction (V2I)	20
1.4.5 Системи на основі GPS систем	21
1.4.6 Системи на основі комп'ютерного зору	21
1.4.7 Системи на основі технології RFID	22
1.5 Основні типи сенсорів	23
1.5.1 Пасивний інфрачервоний датчик	23
1.5.2 Ультразвуковий зворотний паркувальний датчик.....	23
1.5.3 Електромагнітний датчик паркування	23
1.5.4 Активний інфрачервоний датчик	24
1.5.5 Індуктивні детектори	24
1.5.6 Магнітометр	24
1.5.7 П'єзоелектричний датчик.....	24
1.5.8 Пневматична дорожня труба.....	25
1.5.9 Мікрохвильовий радар	25
1.6 Виклики впровадження системи Smart Parking	25
1.7 Висновки.....	26
2 ІСНУЮЧІ ПРОТОТИПИ РОЗУМНИХ ПАРКОВОК.....	27
2.1 Аналіз існуючих прототипів розумних парковок	27
2.2 Структура прототипу системи Smart Parking	32
2.2.1 Сенсорний рівень.....	33
2.2.2 Мережевий рівень.....	34
2.2.3 Проміжний рівень.....	34

	12
2.2.4 Прикладний рівень	35
2.3 Послуги прототипу системи Smart Parking.....	35
2.4 Модель оптимального розміщення датчиків	38
2.5 Висновки.....	42
3 РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ	43
3.1 Структура проекту.....	44
3.1.1 Структура настільного застосунка	44
3.1.2 Структура бібліотеки для обчислень.....	45
3.2 Тестування програми.....	55
3.3 Висновки.....	60
4 РОЗРОБЛЕННЯ СТАРТАП-ПРОЕКТУ “SMART PARKING COVERAGE”	61
4.1 Опис ідеї проекту	62
4.2 Технологічний аудит ідеї проекту.	64
4.3 Аналіз ринкових можливостей запуску стартап-проекту.....	65
4.4 Розробка ринкової стратегії проекту.....	75
4.5 Розробка маркетингової програми	79
4.6 Висновки.....	84
ВИСНОВКИ.....	86
ПЕРЕЛІК ПОСИЛАНЬ.....	88

Перелік умовних позначень

IoT – Internet of Things

WSN – Wireless sensor networks

WPF – Windows Presentation Foundation

XAML – eXtensible Application Markup Language

RFID – Radio Frequency IDentification

FLC – Fuzzy logic controller

ПЛІС – Програмована логічна інтегральна схема

ARM – Advanced RISC Machine

ВСТУП

На сьогоднішніх автостоянках немає єдиної системи для моніторингу паркувальних місць. Існуючі системи сильно залежать від людини. Це призводить до втрат часу, надмірного виділення газів, а також нерівномірності заповнень зон для паркування. Для подолання таких методів були запропоновані різні методи проблеми, такі як служби коротких повідомлень, ZigBee, RFID тощо. У нещодавньому опитуванні дослідники виявили, що в середньому за один рік на пошуки парковки спалюється близько 170 тисяч літрів пального і виробляється 730 тон CO₂. Отже, при використанні розумної системи паркування, яка надаватиме можливість для резервування місць паркування чи пошуку оптимально близької парковки, можна значно скоротити витрати пального, що витрачається «на вітер» при пошуку місця. Бронювання можна здійснювати за допомогою багато платформних інструментів, таких як додаток для Android, додаток для iOS, програми для Windows або навіть веб-портал, що дозволяє користувачеві без проблем знайти собі паркомісце.

Дамо визначення системи Smart Parking. Smart Parking - це типова складова розумної міської програми, розумна парковка є гарною ілюстрацією того, як зробити Інтернет речей (IoT) частиною нашого повсякденного життя, щоб моментальний доступ до інформації в режимі реального часу. Інтернет речей може бути використаний, наприклад, для надання водієві доступу до системи Smart Parking через мобільний додаток, щоб знайти та забронювати місце для стоянки в будь-якій зоні для парковки. Це надає можливість зробити оплату чи передоплату за місце для паркування за допомогою кредитної картки. Система Smart Parking дозволяє знайти вільні парковки, повідомити про ймовірність того, що місце стоянки є ще доступним та приймає рішення про резервування та передоплату за таке місце для паркування. Одним із варіантів реалізації системи Smart Parking є

моніторингу стоянки, де кожен паркувальний пункт обладнаний датчиком (фотоапарат або датчик наявності) для виявлення наявності / відсутності транспортних засобів з метою створення карти наявності, яка може бути використана для керування паркуванням, резервуванням та іншими послугам.

Важливим аспектом при розгортанні системи Smart Parking є побудова оптимальної топології сенсорної бездротової мережі на парковці. Економія коштів на обладнання для сенсорних бездротових мереж дозволить привернути увагу інвесторів, власників торгових центрів, власників автостоянок чи державні служби до системи Smart Parking. Також важливим є зменшення об'єму роботи для мережевих інженерів, які зможуть автоматизувати процес створення топології бездротової мережі на парковці. Тобто програма, що буде розраховувати оптимальну топологію бездротової мережі буде економити не лише кошти на обладнання для мережі, а й звільнить від цієї задачі мережевих інженерів, що є безумовним плюсом. Також програма дозволить змінювати конфігурацію приладів для бездротової мережі а також розробляти та завантажувати до неї будь-які схеми парковок, що описані в правильному форматі у вигляді JSON чи XML файлу. Це дозволить використовувати програму для будь-яких мережевих приладів та різних конфігурацій парковки.

1 СИСТЕМИ SMART PARKING

1.1 Визначення

Система Smart Parking - це типова складова розумної міської програми, розумна парковка є гарною ілюстрацією того, як зробити Інтернет речей (IoT) частиною нашого повсякденного життя, щоб моментальний доступ до інформації в режимі реального часу. Інтернет речей може бути використаний, наприклад, для надання водієві доступу до системи Smart Parking через мобільний додаток, щоб знайти та забронювати місце для стоянки в будь-якій зоні для парковки. Це надає можливість зробити оплату чи передоплату за місце для паркування за допомогою кредитної картки. Система Smart Parking дозволяє знайти вільні парковки, повідомити про ймовірність того, що місце стоянки є ще доступним та приймає рішення про резервування та передоплату за таке місце для паркування. Одним із варіантів реалізації системи Smart Parking є моніторингу стоянки, де кожен паркувальний пункт обладнаний датчиком (фотоапарат або датчик наявності) для виявлення наявності / відсутності транспортних засобів з метою створення карти наявності, яка може бути використана для керування паркуванням, резервуванням та іншими послугам.

1.2 Історія Smart Parking

Здається, що слово "smartness" або "smart" має різні значення відповідно до вимог людей і часу. Протягом багатьох років кількість технологічних паркувальних систем зростає. Автомобільні паркувальні системи у різних проявах були ще з винайдення автомобіля. У будь-який час, скрізь де спостерігався значний обсяг трафіку, стояли автостоянки. Автомобільні паркувальні системи були вперше розроблені на початку 20 століття у відповідь на необхідність зберігання транспортних засобів. Дійсно, використання електронних інтелектуальних систем паркування почалося ще в 1920-х роках, оскільки в американських містах, таких як Лос-Анджелес, Чикаго, Нью-Йорк з'явилися автоматизовані системи паркування. Окрім того, в Нью-Йорку в одній з автостоянок у арт-деко. Система, яка поширена по всій Японії це "Ferris wheel" або "paternoster system", що була створена корпорацією Westinghouse в 1923 і реалізована в 1932 році на вулиці Монро в Чикаго. Протягом останніх двох десятиліть концепція інтелекту з точки зору інтелектуальних паркувальних систем стала більш популярною в найбільш популярних та густонаселених містах, особливо в торгових і розважальних центрах. У середині 80-х років системи, що використовуються для паркування, базувалися головним чином на традиційному методі натискання кнопки на пристрої поруч з контрольно-пропускним пунктом, щоб отримати квиток на паркування та на виході, водій повинен сплатити, перш ніж покинути парковку. Це був метод, який використовувався для визначення, скільки машин входило і виходило до системи щодня, і це було використано для підрахунку кількості доступних вільних місць. Цей варіант паркувальної системи почався з використання різних методів, такі як датчики або бар'єри, щоб мати можливість дізнатися стан автостоянок. Всі ці методи до теперішнього часу суттєво розвинули термін "розумне міське бачення".

1.3 Основні проблеми впровадження Smart Parking

Є основні проблеми, з якими щодня стикаються сучасні транспортні системи та водії, спеціальні паркувальні системи, для яких повинні бути підготовлені розумні міські інженери та розробники. Численні недавні дослідження привели до висновку, що нові інтелектуальні паркувальні системи потрібні практично в кожному мегаполісі в світі, особливо в найближчі десять років, щоб полегшити багато проблем, таких як споживання бензину і викиди забруднення, а також для поліпшення економії часу та зменшення транспортних пригод чи заторів при пошуку місця для паркування. Тому, щоб будь-яка запропонована система вважалася розумною стосовно процесу паркування, вона повинна відповідати наступним вимогам:

- 1)Здатність точно сприймати кількість автомобілів в режимі реального часу.
- 2)Надавання рекомендацій користувачу щодо наявної парковки.
- 3)Спрощення паркувального процесу.
- 4)Можливість інтелектуальних рішень, використовуючи дані, в тому числі додатки в режимі реального часу та історичні аналітичні звіти.
- 5)Вміти надавати користувачеві всю необхідну інформацію про стан будь-яких змін на парковці, яка може трапитися в реальному часі

Ці проблеми повинні бути вирішені з самого початку, щоб забезпечити ефективну роботу системи. Багато досліджень, пов'язані із традиційними системами інтелектуальної парковки в останнє десятиліття, показали, що вони не задовольняють вимог водіїв і бюджет паркувальної техніки.

1.4 Класифікація систем інтелектуальної парковки

1.4.1 Системи на основі агентної моделі (agent model)

Системи, засновані на цій техніці, були запропоновані в академічній спільноті. Ці типи систем можуть бути будь-якими суб'єктами, здатними відстежувати зміни через датчики, як система, що реагує на зміни навколишнього середовища шляхом обміну інформацією та взаємодією з ним. Основні характеристики цих систем це автономність, реактивність та адаптованість. По суті, мультиагентне моделювання - це метод моделювання, розроблений для опису систем з сутностями, автономністю та взаємодією. Інтелектуальні паркувальні системи на базі мультиагентних систем є однією з форм технологій агентних систем з багатопроцесорною системою.

На основі цієї моделі існує метод багатофазної навігації на основі дворівневої карти трафіку, що використовується для узгодження маршруту паркування та керівництва. Підхід розподілу передбачає побудову активної рекомендаційної системи керування паркуванням (APGIS). APGIS складається з машин, автопарків та інформаційно-сервісних центрів парковки (PISC), який має чотири функції: пошук місця для стоянки, вартість паркування, резервування місць на стоянці та узгодження маршрутів паркування.

1.4.2 Системи на основі нечіткої логіки

З 1965 року, коли нечітка логіка була представлена професором Заде, вона зіграла видатну роль в розробці, виробництві, промисловості. Насправді, нечіткі системи управління є системами керування на основі нечітких логічних систем, які аналізують аналогові вхідні значення як змінні, які приймають значення від 0 до 1, тоді як цифрова логіка діє на дискретних значеннях 1 або 0. На сьогоднішній день

нечітка логіка є стандартною технологією, яка застосовується в аналізі даних та сигналів. Нечіткий код розроблений контролювати щось, як правило, щось механічне. В 2006 році запропонована система, яка залежить від нечіткого логічного контролера (FLC). Перевага використання FLC на основі ПЛІС в порівнянні з програмним забезпеченням FLC полягає в тому, що потрібно менше часу на обробку інформації. По-перше, вибрано систему нечіткого керування. Потім, реалізація правил, заснованих на архітектурі нейронної мережі. Це дає змогу навчатись та адаптуватися до певних вимог. Радіо-хвильовий передавач і антена можуть надавати інформацію про наявність місць на парковці і надсилати цю інформацію клієнтам і адміністраторам об'єктів.

1.4.3 Системи на основі бездротових сенсорних мереж

Ці типи систем викликали підвищений інтерес дослідників з 2005 року. Це найбільш популярна технологія в останнє десятиліття, оскільки бездротові сенсорні мережі мають різні переваги, такі як гнучкість, інтелект, дешевизна, швидке розгортання та відстеження. Цей тип системи, який використовує датчики для моніторингу умов навколишнього середовища, широко використовуються, особливо в навчальних закладах, завдяки простоті налаштування та невеликій ціні.

1.4.4 Системи на основі Vehicle to Infrastructure interaction (V2I)

Взаємодія автомобіля з інфраструктурою, подібно до V2V, заснована на бездротових технологіях зв'язку. Комунікація V2I (зазвичай називається V2X) також є широко досліджуваною темою в Сполучених Штатах. Основна ціль це безпека руху. V2I - бездротовий обмін критичними безпечними та експлуатаційними даними між транспортними засобами та інфраструктурою доріг,

спрямованих, перш за все, на те, щоб уникнути або пом'якшити аварії автомобільного транспорту, а також забезпечити широкий спектр інших можливостей безпеки, мобільності та екологічних переваг. Комунікації V2I застосовуються до всіх типів транспортних засобів та всіх доріг, а інфраструктурне обладнання перетворюється на "інтелектуальну інфраструктуру". Вони включають алгоритми, які використовують дані транспортних засобів та елементи інфраструктури, для виконання розрахунків, які попередньо визначають стан ситуації, в результаті чого з'являються попередження та рекомендації для водія. Однією особливо важливою перевагою є здатність систем сигналізації дорожнього руху передавати інформацію про фазу та часові сигнали (SPAT) в транспортний засіб, що забезпечує доставку активних рекомендацій з безпеки та попереджень водіям.

1.4.5 Системи на основі GPS систем

Технологія глобального позиціонування (GPS) використовується для визначення та відстеження точного місця розташування автомобіля. У цій галузі він використовується для надання інформації про місцезнаходження та доступність паркувальних місць у пункті призначення. Ця методика запропонована в 2002 році.

1.4.6 Системи на основі комп'ютерного зору

Останнім часом багато дослідників зосереджували увагу на цих методах. Ця галузь включає в себе методи отримання, обробки, і аналізу зображень. Він використовує комп'ютери для імітації людського бачення, включаючи навчання та вміння робити висновки та вживати заходів на основі візуальних входів, також цю технологію називають комп'ютерним зором. Мета комп'ютерного зору полягає в

тому, щоб комп'ютери ефективно сприймали та обробляли візуальні дані, такі як зображення та відео, а також реагували на зміну цих зображень. Зазвичай, техніка передбачає аналіз кількох кадрів в секунду, а потім посиляє дані в центральну базу даних, після чого користувач може отримати інформацію про зміни в автостоянці. У 2004 році була розроблена система, що використовувала відео нагляд для виявлення наявності автомобіля або транспортного засобу на певній автостоянці. Певна кількість пікселів у градаціях сірого використовується як порогові значення для відрізняти пікселі від автомобіля та від незайнятого місця. Інша система паркування, яка називається CCTV використовує зображення для виявлення паркувальних місць. Камери відеоспостереження встановлюються на автостоянки для автоматичного виявлення місця для паркування. Однак ці методи можуть неправильно виявляти припарковані транспортні засоби. Система націлена на випадки, коли необхідні значення заповнюваності.

1.4.7 Системи на основі технології RFID

У багатьох наукових працях рішення RFID для смарт-паркування дозволяють керувати легко проводити моніторинг парковку, особливо на етапах прототипу. Основний механізм RFID-технології залежить від електромагнітного поля для ідентифікації та відстеження міток, доданих до об'єктів.

1.5 Основні типи сенсорів

1.5.1 Пасивний інфрачервоний датчик

Пасивні інфрачервоні датчики визначають стан зайнятості паркувального місця, виявляючи зміну випромінення, що продукується автомобілем та дорогою. Чутливість датчика зменшується при сильному дощі, снігу чи густому тумані.

1.5.2 Ультразвуковий зворотний паркувальний датчик

Ультразвуковий зворотний паркувальний датчик швидко стає застарілим на ринку сенсорів. Він в основному складається з чотирьох або більше датчиків, прикріплених до заднього бампера вашого автомобіля.

1.5.3 Електромагнітний датчик паркування

Електромагнітний датчик паркування є більш обережним, ніж його примітивний аналог. Він встановлений позаду автомобільного бампера, можна зробити його практично невидимим. Він також більш надійний, ніж ультразвуковий датчик паркування; має змогу виявляти об'єкти з боків на відміну від ультразвукової технології, яка виявляє об'єкти безпосередньо позаду вашого бампера. Навіть якщо цей сенсор коштує дорожче, ніж ультразвуковий датчик, він значно надійніший та інформативніший.

1.5.4 Активний інфрачервоний датчик

Активний інфрачервоний датчик виявляє транспортні засоби, випромінюючи через ІЧ-порт сигнал і потім приймає ехо-сигнал. Активні інфрачервоні датчики використовуються для проведення декількох операцій. Точне вимірювання положення транспортного засобу(швидкість і клас), передача даних від датчика Цей сенсор чутливий до умов навколишнього середовища.

1.5.5 Індуктивні детектори

Це провідні ланцюги різного типу і розміри, які працюють із сигналами частотою від 10 до 50 кГц. Частота коливань індуктивного кола безпосередньо регулюється індуктивністю петлі, яка змінюється з присутністю автомобіля.

1.5.6 Магнітометр

Магнітометр є двох типів: з флюсом і індукційний або пошуковий спіральний магнітометр. Магнітометри нечутливі до погодних умов. Індукційні або пошукові спіральні магнітометри також нечутливі до погодних умов.

1.5.7 П'єзоелектричний датчик

П'єзоелектричні датчики перетворюють кінетичну енергію на електричний заряд, коли спеціально оброблений матеріал піддається вібрації або механічному впливу. Це дає більш точність, але потребує декількох детекторів. Вони чутливі перепадів температур.

1.5.8 Пневматична дорожня труба

Транспортні засоби виявляються шляхом створення повітряного тиску, який закриває перемикач і виробляє сигнал, коли транспортний засіб пройде над ним. Датчик недорогий сам по собі та в експлуатації проте чутливий до температури та погодних умов.

1.5.9 Мікрохвильовий радар

Цей датчик передає сигнал через антену та виявляє автомобіль за ехо сигналом, відбитим назад до антени. Два типи радарних датчиків - безперервний радар (CW), а також частотно-модульований радар неперервної хвилі (FMCW). Вони нечутливі до погодних умов, але доплерівські сенсори повинні бути оснащені допоміжними датчиками в спробі виявити нерухомий транспортний засіб.

1.6 Складності впровадження системи Smart Parking

В Розробка та впровадження системи розумної парковки потребує вирішення декількох проблем:

- Вибір сенсора: Вибір відповідної технології для моніторингу є складною проблемою що залежить від ефективності розгортання розумної парковки. Розмір, надійність, адаптація до змін навколишнього середовища, надійність та вартість є одними з параметрів, які відіграють ключову роль при виборі типу датчиків для розумної парковки.
- Проблеми з протоколом. Необхідно використати загальновідомий протокол, для зручного розширення та модернізації системи.

- Питання енергозабезпечення. Ще однією проблемою є підключення живлення для всіх датчиків з ефективним енергоспоживанням.
- Потенційні недоліки. При розгортанні на вулиці, ультразвукові датчики, як відомо, чутливі до зміни температури та турбулентності. Потенційні недоліки, пов'язані з бездротовим зв'язок, такі як інтерференція, відбиття, дифракція та проникнення в стіну, є викликами, які доведеться вирішувати при розгортанні Smart Parking в приміщенні.
- Ефективне бездротове охоплення: для внутрішнього та зовнішнього розгортання. Ефективність покриття бездротової мережі є важливим параметром, від якого залежить час реакції інтелектуальної системи паркування.

1.7 Висновки

В цьому розділі було розглянуто загальну проблематику систем Smart Parking, основні проблеми при впровадженні та реалізації цієї системи, основні типи систем, проаналізовано елементну базу сенсорів, розглянуто можливості та обмеження різних сенсорів. Також було розглянуто проблеми що можуть виникнути при впровадженні системи Smart Parking такі як: вибір сенсора, відсутність єдиного стандартизованого протоколу, енергоживлення сенсорів, вплив природних явищ на коректність роботи системи.

2 ІСНУЮЧІ ПРОТОТИПИ РОЗУМНИХ ПАРКОВОК

2.1 Аналіз існуючих прототипів розумних парковок

Управління паркуванням було розглянуто з використанням двох моделей: контроль парковки і парковка множинного моніторингу. Ґрунтуючись на обліку записів входу і виходу зі стоянки, інтелектуальне управління паркуванням на основі моніторингу воріт може надавати важливі послуги для водіїв, такі як можливість перевірки наявності вільного місця на парковці і резервування через Інтернет. У розумному управлінні паркуванням на основі моніторингу, кожне місце паркування обладнано датчиком (датчик з камерою або датчик присутності) для виявлення наявності або відсутності транспортних засобів з метою створення карти доступності, дана карта може бути використана для управління паркуванням, резервування та інших послуг, описаних вище, в поєднанні з RFID. У той час як модель моніторингу воріт призводить до створення простих і недорогих паркувальних систем, модель управління дорожча, але надає набагато більше послуг для водія і власника автостоянки. Інтелектуальне управління паркуванням, що використовує моніторинг паркомісць, можна розділити на управління з декількома парковками, що використовується для управління різними автостоянками в різних приміщеннях і на відкритому просторі або в одній парковці, управління в основному призначений для паркування на стоянці і фокусується на управлінні однією парковкою. [3].

Автори [1-3] пропонують розумних менеджерів парковки з використанням технології RFID і вбудованих систем з метою допомогти користувачеві перевірити доступність паркувального місця через Інтернет. Автори [4,5] пропонують систему, яка використовує виявлення зображення. В [6-8] описується бездротова сенсорна

мережа з бездротовим датчиком, де бездротові датчики розміщуються на автостоянці на кожному паркувальному місці, забезпеченим одним сенсорним вузлом. Автори [9,10] пропонують систему з декількома паркувальними системами для розширення послуг для масштабування, більш, ніж одна автостоянка; наприклад, управління паркуванням на відкритому просторі, наприклад, на вулицях, що прямують до відповідної автостоянки в місті. Це може започаткувати співпрацю між декількома парковками.

В [11] автори пропонують KATHODIGOS, яка представляє собою інтелектуальну систему паркування, яка використовує бездротові технології та бездротові додатки для отримання інформації про стан парковки на дорозі. Отримана інформація передається в центральну інформаційну систему.

У рішенні [12] пропонується структура управління паркуванням під назвою iParking, яка контролює входні / вихідні транспортні засоби з використанням сенсорної мережі. iParking обчислює кількість доступних місць для паркування та поширює інформацію на клієнтів автостоянки.

В [13] автори описують систему паркування вулиць на основі бездротових сенсорних мереж. Система може контролювати стан кожного паркувального місця, розгортаючи магнітний сенсорний вузол на просторі.

Пропонується алгоритм виявлення транспортних засобів для точного виявлення паркувального автомобіля, а адаптивна вибірка використовується для зниження споживання енергії. «Nawaz et al» [14] представляє сенсорну систему на основі смартфона, яка виявляє водія, що звільняє місце для парковки.

Більшість пропонованих рішень до сих пір розглядають використання WSN або RFID і фокусуються на проектуванні транспортного засобу, що дозволяє відмовитися від проектування мережі за рахунок ефективного розміщення датчиків на стоянці. Починаючи з великого обсягу досліджень, присвячених проблемі розташування базової станції (BS), оптимальне розміщення вузлів в сітчастої

мережі розвинулося в напрямку оптимізації топології бездротової мережі з рішеннями від динамічного програмування [15] до генетичних алгоритмів [16,17].

Запозичена з області досліджень операцій проблема розташування BS є частиною просторового розташування об'єкта, де набір точок попиту повинен охоплюватися набором об'єктів з метою оптимізації заданої мети, такої як загальна відстань між точками попиту і їх найближчими об'єктами, з деякими обмеженнями.

Це тісно пов'язано з проблемою максимального покриття (MCLP) [19,20], також званою проблемою розподілу розташування в WSN, де максимально можливу кількість точок попиту має бути покрито p -датчиками фіксованого радіуса.

Розглядаючи технічні питання WSN для побудови IoT, дослідники запропонували кілька рішень для досягнення оптимального розміщення вузлів за допомогою управління топологією.

Роботи з максимізації покриття мережі з невеликими вимогами до зв'язку між датчиками або їх відсутністю представлені в [21-26], аналогічні завданням визначення місця розташування BS, оскільки вони спрямовані на оптимізацію розташування датчиків, щоб максимізувати їх колективне покриття.

У той час як [22] використовувало цілочисельне програмування, роботи в [21,23-25] засновані на різних евристичних правилах для поступового розгортання датчиків. Техніка, яка використовується в [26], адаптована з методів віртуалізації для розгортання датчика.

Шляхом виявлення оптимальних способів підключення датчиків за допомогою вирішення проблеми оптимального розміщення, оптимізації розкладки, наприклад, наведені в [27-31], можуть також знизити споживання енергії.

У той час як [27,29] засновані на евристичному вирішенні з використанням багатоцільової еволюційної оптимізації, проблема розміщення в [31] сформульована як проблема нелінійної оптимізації використовуючи саме

інкрементний алгоритм, який додає вузли по черзі в мережу найбільш ефективним чином.

В [30] оптимальне розміщення статичних датчиків в мережі використовується, щоб допомогти агенту переміщатися в області, використовуючи вимірювання діапазону для датчиків, щоб локалізувати себе.

Прагнучи вивчити різні методи правильної сітки для сенсорних мереж, [32] виводить в кожному випадку (квадратне, трикутне і шестикутне) мінімальну кількість датчиків, необхідних для забезпечення повного охоплення, і отримана похибка виражається в мінімальному числі вузлів, які необхідно відключити, щоб знизити охоплення мережі.

Дивлячись на позиціонування датчиків в місцевості з точки зору передачі даних, робота, представлена в [33], ділить досліджуваний ландшафт на клітини, а потім аналізує, як вони повинні розподілятися між осередками таким чином, щоб уникнути мережевих вузьких місць і втрати даних.

В [34] представлені два жадібних алгоритму, які обирають місця для мережі датчиків з мінімальною кількістю вузлів з використанням сітчастої моделі для місцевості та ймовірнісної моделлю охоплення датчиків.

Модель, заснована на просторових обмеженнях, дозволяє включити вплив перешкод і висоти місцевості і включення фактору важливості, віддаючи перевагу покриттю деякої частини ландшафту.

У той час як велика кількість зусиль було інвестовано в інтелектуальне управління паркувальними місцями з особливим наголосом на проектування транспортних засобів з показаннями датчиків від паркувальних місць до шлюзу, оптимальне розміщення сенсорних пристроїв на стоянці для збільшення охоплення та підключення економічним способом є важливим параметром, від якого залежить ефективне управління паркуванням.

Розглянемо проблему розміщення вузлів в бездротових сенсорних мережах з метою максимізації охоплення і терміну служби мережі при плануванні мережі сітчастих датчиків для інтелектуальної системи паркування.

Використовуючи математичне формулювання, засноване на цілочисельному лінійному програмуванні, пропонуються однокрокові і двокрокові рішення, реалізовані на мові Мозеля на основі бібліотек XPress-MPsuite. Використовуючи оптимізацію Мозеля, запропоновано удосконалення реалізації, які дозволяють знайти найкращу кількість вузлів, які оптимізують цільову функцію в мінімальний час обчислення.

Це рішення згадується в роботі як «оптимізоване» рішення в порівнянні з «статичним» рішенням, яке знаходить рішення в більш складних обчисленнях.

Експериментальні результати показують відносну ефективність одноступінчастої в порівнянні з двоступеневої моделлю за різними параметрами продуктивності. На відміну від багатьох досліджень, в яких використовуються спрощені моделі, засновані тільки на діапазоні зв'язку для оптимального розміщення датчиків в чутливому середовищі, модель об'єднує діапазон чутливості як важливий параметр, який може впливати не тільки на інженерні, але і на економічна ефективність інтелектуальної системи паркування .

Розглянута модель інтелектуальної парковки заснована на прототипі реальної парковки, запропонованому в [35], який був протестований на відкритій автостоянці в Центрі CERISTresearch в Алжирі, Алжир.

2.2 Структура прототипу системи Smart Parking

Інтелектуальна система паркування (SPS), була побудована на основі методу датчиків і розроблена на основі багатошарової структури, що забезпечує модульність і масштабованість, а також надання різних послуг різним користувачам системи паркування.



Рисунок 2.1 – Структура мережі

Структура, зображена на рисунку 2.1, включає в себе чотири рівні: сенсорний, мережевий, проміжне ПО і прикладний рівень.

2.2.1 Сенсорний рівень

Цей шар визначає платформу, на якій сенсорні пристрої вбудовані в автостоянку для виявлення присутності / відсутності автомобіля, а пристрої RFID, розташовані на автомобільних воротах і стратегічні точки парковки, використовуються для ідентифікації автомобілів на основі унікального відображення між мітками RFID і автомобілями.

Використовуються три типи сенсорних пристроїв:

- Підлеглі пристрої, також звані «приймачами», які поміщаються на паркувальні місця для виявлення присутності / відсутності;
- Передавальні пристрої, також звані «передавачі», яким доручено збирати показники датчиків зі своїх підключених підлеглих пристроїв і передавати ці дані на комутатор для подальшої обробки;
- «Якірні» пристрої, що використовуються в якості комутаторів, для збільшення зони охоплення парковки та ефективної маршрутизації показань датчика.

Ведені пристрої підключаються до провідних пристроїв за допомогою дротового зв'язку з використанням I2Cprotocol. Для цілей вимірювання використовувалися мікроконтролери, обладнані ультразвуковими датчиками, а в якості шлюзу використовувалася плата Panda на основі ARM.

2.2.2 Мережевий рівень

У цьому рівні були запропоновані різні способи зв'язку для підтримки зв'язку від ведучого і якірних датчиків до шлюзу і від шлюзу до користувачів парковки (водіїв, віддалених користувачів і власників парковки).

До цього рівня відносяться:

- Зв'язок 802.15.4 / ZigBee для маршрутизації показань датчиків від провідних датчиків до шлюзу;
- TCP / IP через Ethernet для підключення шлюзу до сервера парковки і бази даних;

Доступ в Інтернет для віддаленого доступу до інтелектуальної системи паркування зовні.

2.2.3 Проміжний рівень

У Рівень в якому розпізнавання ситуацій здійснюється за допомогою інтелектуальних алгоритмів та технологій ефективної візуалізації для представлення сервісам та для зручності користувачів.

Цей рівень містить різні бази даних та пов'язані з ними сервери і керує всією програмною розвідкою, наданою системою розумної парковки, для надання користувачам інтелектуальних послуг, забезпечуючи зв'язок між прикладним рівнем, за яким послуги запитуються, а також нижчих шарів, в яких інтелектуальні пристрої вбудовані в автостоянку для надання інформаційних послуг.

2.2.4 Прикладний рівень

Прикладний рівень - це рівень, в якому різні служби визначаються та надаються різним користувачам. Клієнтські пристрої підключені через протокол TCP / IP до бази даних паркування. Останнє оновлюється в режимі реального часу за статусом автостоянок. Для моніторингу стоянки було розглянуто два види клієнтських додатків:

- Додаток для мобільних пристроїв для телефонів і планшетів;
- Настільний додаток для ноутбуків та комп'ютерів.

2.3 Послуги прототипу системи Smart Parking

Послуги, надані системою розумної парковки, включають:

- Активний сервіс: у поточній реалізації активна RFID-технологія була використана для заміни паперової системи продажу квитків та надання інструкцій водіям, надаючи водіям автомобілів можливість використовувати свій тег RFID, щоб керувати парковкою. У цьому випадку зчитувачі RFID, розміщені в стратегічних точках автостоянки, використовуються для отримання ідентифікатора мітки та для опитування бази даних, якими керує диспетчер стоянки, щоб отримати попередньо розподілене місце для паркування. Передбачається, що кожен автовласник отримає активну позначку RFID.
- Служба пошуку автомобілів: ця служба також використовує RFID, що перевозиться водієм для опитування зчитувачів, розташованих у різних місцях, та передачі ідентифікатора тегу адміністратору паркування. Останній перевіряє місця зайнятої парковки в базі даних та на стоянці та повертає

відповідь до вузла вказівки для паркування для візуалізації за допомогою системи керування візуалізацією (VMS).

- Інтелектуальне регулювання паркування: регулювання розумної парковки відбувається, коли водіям автомобілів встановлена спеціальна автостоянка, попередньо призначена для їх використання, або коли здійснюється резервування місць паркування. Коли машина виявляється на місці паркування, учасник події, включаючи ідентифікатор тегу RFID автомобіля та його місцезнаходження, передається диспетчеру паркування. Останній використовує отриману інформацію для перевірки у своїй базі даних, якщо ідентифікована машина має право на доступ до місцезнаходження.
- Послуги з охорони паркування: поточна реалізація Smart Parking дозволяє водієві надавати або постійний тег RFID у випадку приватного місця для паркування або тимчасового входу в паркінг у випадку громадського місця для паркування. Цей тег використовується для різних цілей, включаючи цілі безпеки, за допомогою: взаємозв'язку між тегамі RFID, що ідентифікують машину та місця для паркування, та процес перевірки безпеки, де тег використовується для виявлення автомобільної крадіжки, коли програмне забезпечення для керування паркуванням виявляє машину, що залишає місце стоянки, тоді як відповідний тег сигналізує найближчому зчитувачу, що знаходиться далеко від визначеного місця для паркування.
- Статус доступності автостоянки шляхом інтеграції в систему виявлення автомобіля джерела світла на місцях паркування, які контролюються виконавчими механізмами, щоб повідомити про стан стоянки: наприклад, червоний - для зайнятого, зелений для порожнього, жовтий - для зарезервованого.
- Обслуговування паркування для заміни паперового платежу за допомогою ідентифікаторів RFID, які збираються на вході в паркінг і використовуються

при виїзді з паркування на платформі RFID на основі оплати, щоб обчислити час, витрачений на стоянці, а потім сплатити рахунок водієві.

- Управління автостоянкою для автоматичного відкриття воріт, щоб підтвердити доступ для водіїв до певної частини стоянки. Ця послуга схожа на сервіс Smart Parking, оскільки вони побудовані навколо системи авторизації, яка спрямована на обмеження доступу на основі дозволів.
- Перевірка дистанційної доступності через Інтернет та / або мережу GSM для моніторингу в режимі реального часу. Забезпечуючи доступ до статусу наявності місць для паркування в будь-який час та в будь-якому місці, ця послуга може забезпечити значну економію часу для користувачів.

До основних проблем при впровадженні цієї системи можна віднести:

- Розробка та впровадження розумних паркувальних систем, описана вище, спричинила ряд проблем. Деякі з них були розглянуті під час реалізації прототипу в CERIST.
- Вибір сенсора: Вибір відповідної технології зйомки є складною проблемою, від якої залежить ефективність розумної парковки. Розмір, надійність, адаптація до змін у навколишньому середовищі, надійність та вартість є одним з параметрів, які зазвичай відіграють ключову роль при виборі типу датчиків для інтелектуальної системи паркування. Крім того, нав'язливі датчики, які потрібно встановлювати безпосередньо на поверхню тротуару, зазвичай уникаються, оскільки вони мають ряд проблем, включаючи потребу копати та тунелювати під дорожньою поверхнею. Ультразвукові датчики були розглянуті під час нашого розгортання, оскільки вони були невисокими та вимагали простої установки шляхом фіксації на землі. Ультразвукові датчики були розглянуті під час нашого розгортання, оскільки вони були невисокими та вимагали простої установки шляхом фіксації на землі.

- Проблеми з протоколом I2C: одна з основних проблем полягає в тому, щоб змінити I2C для нашої заявки, оскільки вона не призначена для роботи на великих відстанях. Необхідні деякі зміни деяких параметрів I2C, такі як опір та тактова частота.
- Питання енергозабезпечення. Ще однією проблемою було підключення джерела живлення для всіх датчиків з ефективним споживанням електроенергії.
- Потенційні недоліки. Коли розгортається на відкритому повітрі, ультразвукові датчики, як відомо, чутливі до змін температури та надзвичайної турбулентності повітря. Потенційні недоліки, пов'язані з внутрішнім бездротовим зв'язком, такі як перешкоди, відбиття, дифракція та проникнення в стінки, є проблемами, які повинні бути вирішені під час розгортання Smart Parking в приміщенні.
- Ефективне охоплення бездротовим доступом: для внутрішнього та зовнішнього розгортання ефективне бездротове покриття є важливим параметром, від якого залежить час життя інтелектуальної системи паркування. Це основна проблема, що вирішувалася в цій системі.

2.4 Модель оптимального розміщення датчиків

На рисунку 2.2 зображено схему інтелектуальної системи паркування, де датчики організовані на групи неоднорідних пристроїв (ранні датчики, магістральні датчики та ретранслятор / комутатори), які взаємозалежні між собою через провідні лінії, використовуючи протоколи I2C та Ethernet або бездротові зв'язки, що працюють на 802.15 .4 / протокол Zigbee. У такому неоднорідному середовищі підпорядковані вузли сенсора розташовані на місцях паркування для виявлення

присутності автомобіля / відсутності автомобіля та звітності на вузол основного датчика, тоді як вузли головного датчика розташовані на краю автостоянки і надсилають повідомлення про датчик показання до шлюзу, де виконується подальша обробка.

Головні та підпорядковані датчики знаходяться на стоянці на основі геолокаційних обмежень, що може призвести до поганого покриття та нездатності показанням датчиків досягти шлюзу, розташованого біля входу на стоянку. Опорні вузли, інтегровані в систему інтелектуальної парковки, щоб пом'якшити цю проблему, стратегічно розмістивши ці датчики в оптимальних для розрахунку місцях, щоб збільшити охоплення та зв'язок сенсорної мережі. Як розгорнуті на розумній автостоянці, підпорядковані та головні вузли утворюють межі бездротової мережі, а якірні вузли становлять основу мережі сенсорів.

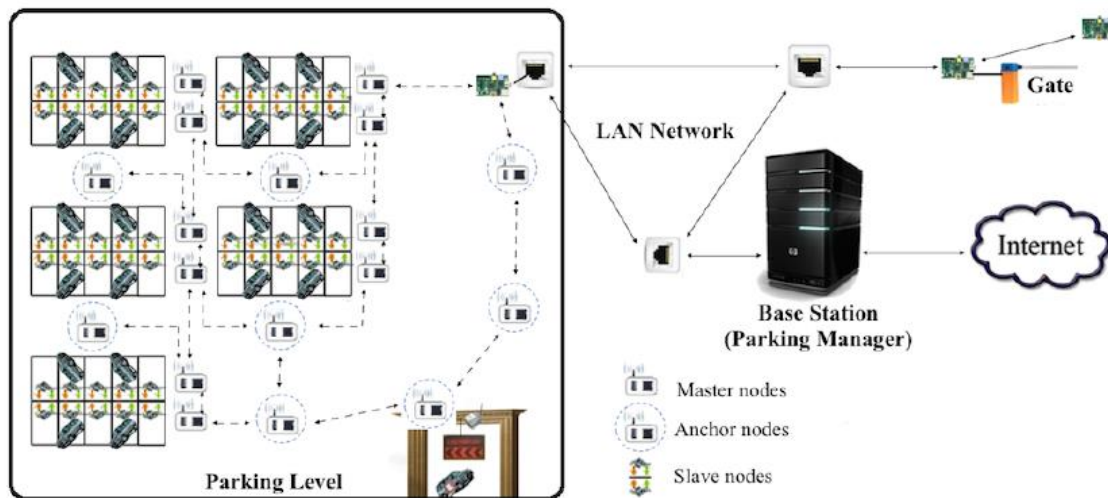


Рисунок 2.2 – Розміщення сенсорів на парковці

Оптимальне розміщення пов'язаних пристроїв - це проблема інженерії в мережі, яка може бути вирішена за допомогою лінійної формули, запропонованої в цьому документі. Покрита площа (парковка) представлена як квадрат $L \times L$ клітин. Ми припускаємо, що будь-яка клітина повністю або частково не покрита, тобто часткове покриття комірки не допускається. Кожен датчик охоплює всі комірки, що

потрапляють в діапазон чутливості і спілкується з усіма іншими датчиками, розташованими в межах діапазону зв'язку. Тривалість життя залежить від відстані між вузлами датчиків і між датчиками та комутатором: чим менше відстань, тим довше тривалість життя і навпаки. Ми також припускаємо, що сенсорна мережа підключена, тобто кожен датчик повинен перебувати в межах діапазону зв'язку іншого датчика, а комутатор повинен перебувати в діапазоні зв'язку щонайменше з одним датчиком. Відстань між двома об'єктами обчислюється як евклідова відстань.

Формулювання моделі вимагає наступного визначення координат кожної комірки. Нехай $N = L \times L$ - кількість доступних клітин. Координати $x_c(i)$ та $y_c(i)$ i -ї клітини, $i = 1; \dots; N$:

$$x_c(i) = \begin{cases} L & \text{if } (i \bmod L) = 0 \\ L \bmod L & \text{otherwise} \end{cases} \quad (1)$$

$$y_c(i) = \begin{cases} i \div L & \text{if } (i \bmod L) = 0 \\ (i \div L) + 1 & \text{otherwise} \end{cases} \quad (2)$$

Отже, евклідова відстань між кожною можливою клітиною:

$$d_{ij} = \sqrt{[x_c(i) - x_c(j)]^2 + [y_c(i) - y_c(j)]^2} \quad (3)$$

Рішення змінних

$$\begin{aligned} x_i & \begin{cases} 1, & \text{если датчик находится в ячейке } i; \\ 0, & \text{иной случай.} \end{cases} \\ s_i & \begin{cases} 1, & \text{если раковина расположена в ячейке } i; \\ 0, & \text{иной случай.} \end{cases} \\ y_i & \begin{cases} 1, & \text{если ячейка } i \text{ покрыта, то она попадает в диапазон чувствительности датчика;} \\ 0, & \text{иной случай.} \end{cases} \\ z_i & \begin{cases} 1, & \text{если датчик находится в ячейке } i, \text{ а другой (или приемник) находится в ячейке } j \\ & \text{, и они находятся в пределах диапазона связи, то есть; } d_{ij} \leq C_R; \\ 0, & \text{если ячейки } i \text{ и } j \text{ находятся в пределах диапазона связи и по меньшей мере одна ячейка пуста.} \end{cases} \end{aligned} \quad (4)$$

Спираючись на вищезазначені позначення та визначення, цільова функція може бути виражена як сума покритих комірок мінус загальна відстань між усіма датчиками і комутаторами:

$$\max \left\{ \sum_{i=1}^N y_i - \sum_{i=1}^N \sum_{j=1}^N d_{ij} \cdot z_{ij} \right\} \quad (5)$$

Набір обмежень для моделі такий:

$$\sum_{i \in N} x_i = N_S \quad (6)$$

$$\sum_{i \in N} s_i = 1 \quad (7)$$

$$x_i + s_i \leq 1 \quad \forall i \in N \quad (8)$$

$$x_i \leq s_i \quad \forall i, j \in N | d_{ij} \leq S_R \quad (9)$$

$$y_i \leq \sum_{j \in N | d_{ij} \leq S_R} x_j \quad \forall i \in N \quad (10)$$

$$x_i + s_i \leq \sum_{j \in N | d_{ij} \leq S_R} x_j \quad \forall i \in N \quad (11)$$

$$z_{ij} \leq x_j \quad \forall i, j \in N | j > i \text{ и } d_{ij} \leq C_R \quad (12)$$

$$z_{ij} \leq x_j + s_j \quad \forall i, j \in N | j > i \text{ и } d_{ij} \leq C_R \quad (13)$$

$$z_{ij} \geq x_j + x_j + s_j - 1 \quad \forall i, j \in N | j > i \text{ и } d_{ij} \leq C_R \quad (14)$$

$$x_i \leq \sum_{j \in N} z_{i,j} \quad \forall i \in N \quad (15)$$

$$s_i \leq \sum_{j \in N} z_{i,j} \quad \forall i \in N \quad (16)$$

Обмеження (6) та (7) гарантують, що всі доступні датчики та комутатор розташовані в зоні розгляду. Обмеження (8) накладає на те, що в кожній клітині може розміщуватися не більше одного датчика або комутатора. Покриття осередку здійснюється за допомогою обмежень (9) та (10). Фактично, якщо в клітині i є датчик, то всі клітини на допустимій відстані (обмеження 9). З іншого боку, якщо клітина покрита, то там повинен існувати датчик на відстані покриття (обмеження (10)). Наступні обмеження складають мережу. Зокрема, якщо датчик або комутатор є в комірниці i , у допустимій дальності (обмеження (11)) повинен бути датчик.

Обмеження (12) - (14) реалізують функцію AND, яка в тому випадку, якщо відстань нижча або рівна допустимій, встановлює зв'язок (тобто $z_{ij} = 1$) тоді і тільки тоді, коли є датчик або комутатор в j . Модель зони покриття зображена на рисунку 2.3.

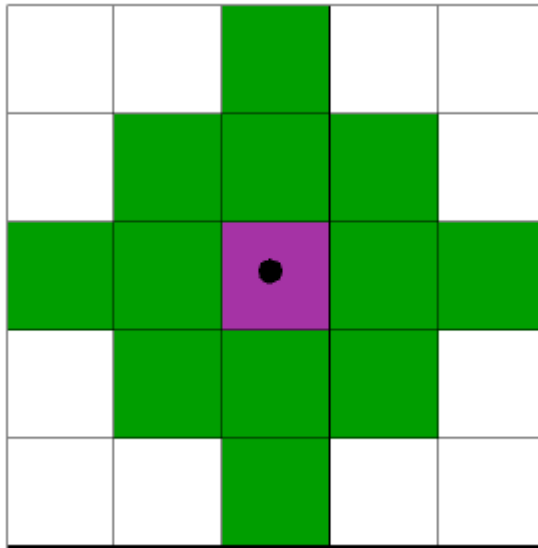


Рисунок 2.3 – модель зони покриття датчика

2.5 Висновки

В цьому розділі було розглянуто алгоритм для оптимізації розміщення сенсорів в системі розумної парковки, описали обмеження та умови що необхідні для даної задачі. Також було розглянуто реальну модель системи Smart Parking на автостоянці в Алжирі. Проаналізовано модель оптимізації, що застосовувалась на реальному прототипі.

3 РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Проаналізувавши існуючі рішення та вивчивши сенсорну базу було прийняте рішення зробити основним об'єктом оптимізації бездротові Bluetooth комутатори, так як для точного визначення присутності автомобіля чи іншого транспортного засобу на парковці необхідно розмістити датчик на кожному місці для паркування. Такий підхід дозволяє точно визначати зайнятість місця при будь-яких погодних умовах та інших завадах.

В цьому розділі буде розглянуто результати роботи програми, засоби для розробки та структуру проекту.

Мовою програмування обрано C#, так як вона поєднує в собі парадигми ООП програмування та надає широкі можливості в організації коду. Обрана мова програмування досягає досягти високого рівня абстракції від реалізації, що робить можливим легко масштабувати проект та вносити в нього певні обчислювані модулі без зміни структури програми.

Основні обчислення вирішено відділити в динамічно приєднувану бібліотеку(DLL), що надає широкі можливості в застосуванні та при інтеграції до інших систем. Це також дозволяє абстрагуватись від інтерфейсу, так як потім можна підключити бібліотеку до будь-якого .Net рішення.

Для розробки інтерфейсу використовується WPF, що розшифровується як «Windows Presentation Foundation», це технологія для розробки і побудови клієнтських застосунків. Ця технологія була обрана через зручність та швидкість розробки з її використанням. У WPF для побудови інтерфейсу використовується мова розмітки XAML, що розшифровується як «eXtensible Application Markup Language». Ця мова розмітки дозволяє швидко розробити інтерфейс для застосунку, також інтуїтивно зрозуміло можна налаштовувати обробники подій.

Основою застосунку будуть слугувати два проекти:

1)Динамічна приєднана бібліотека з основними обчисленнями, яка підключатиметься до проекту з інтерфейсом та надаватиме всі необхідні методи та інтерфейси для взаємодії та відображення необхідної інформації

2)Настільний застосунок для відображення інформації, обчислень та взаємодії з програмою. Застосунок матиме функціонал для завантаження чи вивантаження схеми парковки, обрахунку оптимальної позиції датчиків на парковці.

3.1 Структура проекту

3.1.1 Структура настільного застосунка

Діаграма залежностей застосунка зображена на рисунку 3.1. З діаграми видно, що настільний застосунок складається з чотирьох основних компонентів, а саме:

1MainWindow – відповідає за відображення елементів на формі, обробку подій елементів форми.

2)Settings – це файл в форматі XML в якому зберігаються основні налаштування проекту, такі як версія проекту, версія фреймворку на якому він написаний і інші конфігурації

3)Resources – це файл є фактично файлом з заголовками бібліотек на які посилається проект.

4)App – повністю звучить як App.config, використовується для ініціалізації та збереження конфігураційних даних проекту.

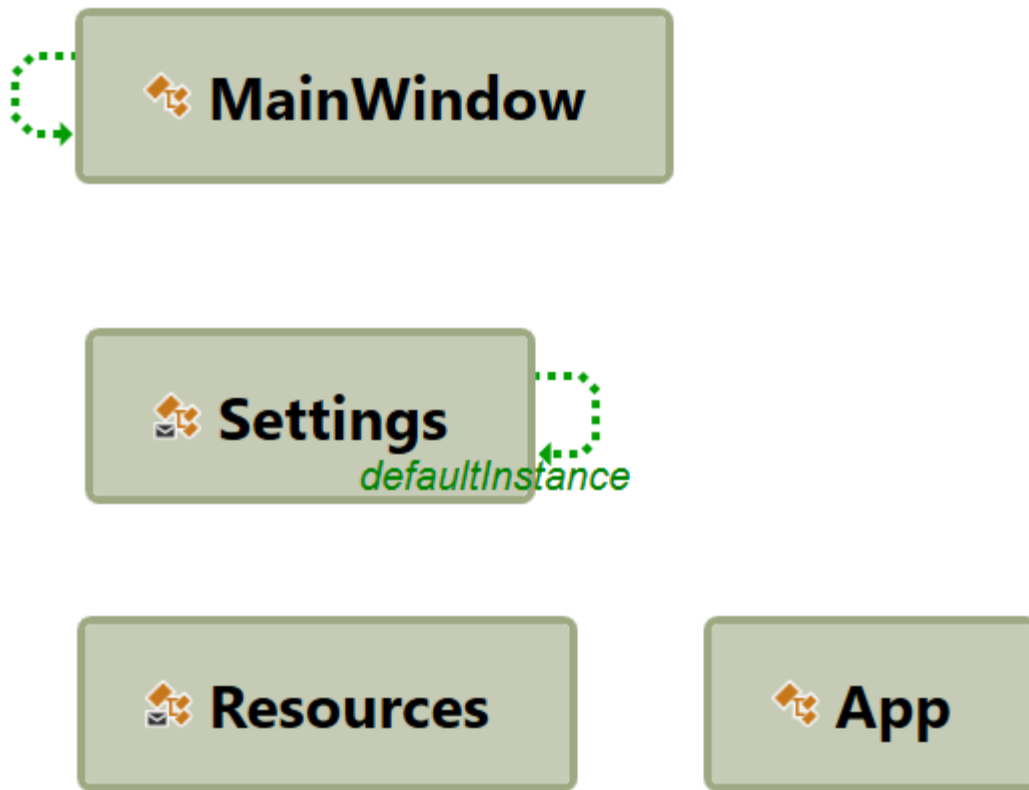


Рисунок 3.1. – Основні складові застосунка

3.1.2 Структура бібліотеки для обчислень

Діаграма залежностей бібліотеки для розрахунків оптимального положення комутуючих пристроїв в мережі зображена на рисунку 3.2

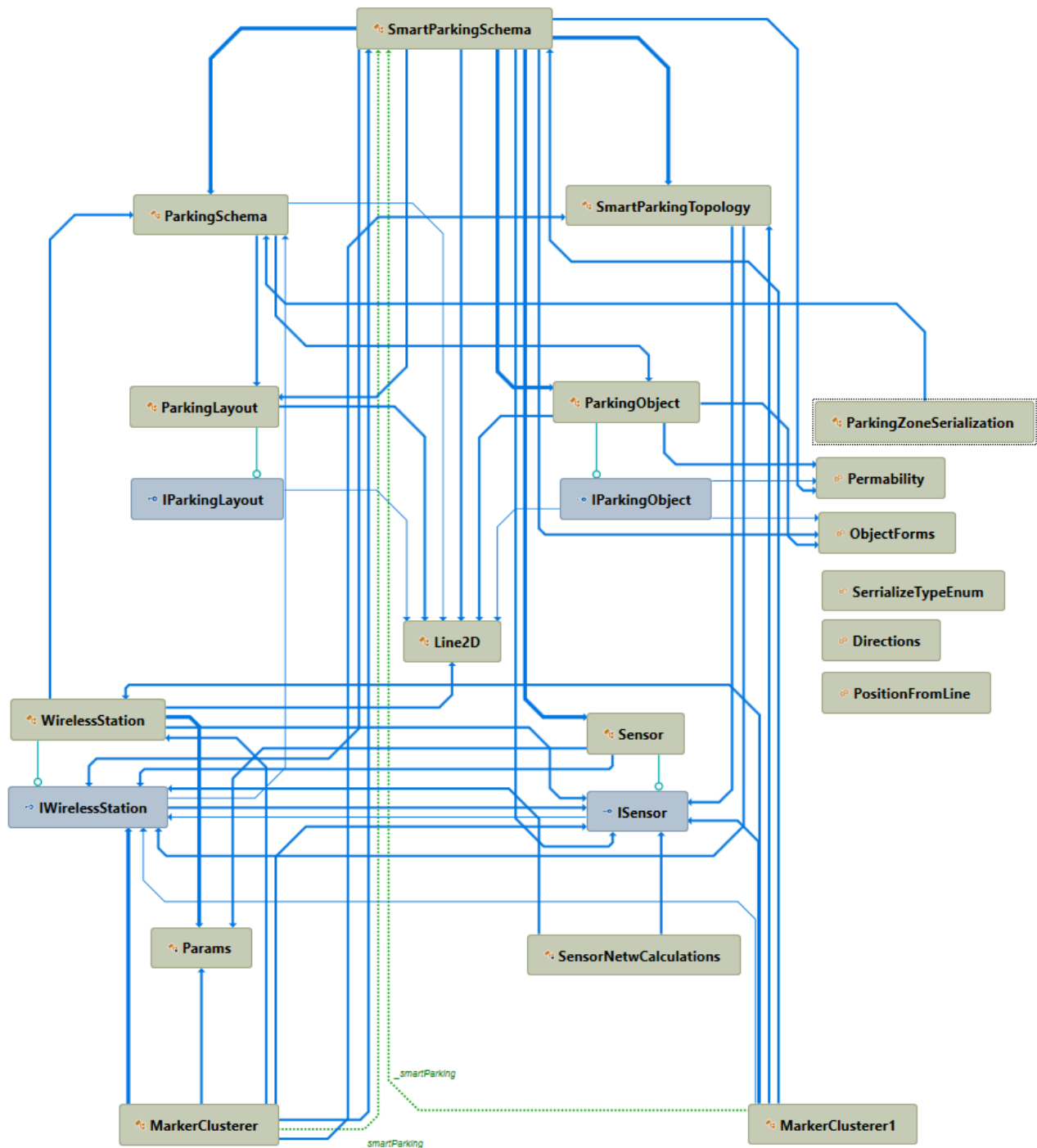


Рисунок 3.2 – Діаграма залежностей бібліотеки

Основним класом є `SmartParkingSchema`. Цей клас зберігає в собі схему парковки(екземпляр класу `ParkingSchema`) і топологію мережі парковки(екземпляр класу `ParkingTopology`). Клас реалізовує 3 методи:

- `GetParkingZoneShape` – метод відповідає за створення графічного відображення для схеми парковки з датчиками на ній.
- `IsSensorConnected` – метод перевіряє чи підключений сенсор до комутатора на парковці, має вхідний параметр екземпляр класу, що реалізовує інтерфейс `ISensor`. Метод проходить по всім бездротовим комутаторам в мережі та перевіряє чи містять вони підключення до цього сенсора:

```
public bool IsSensorConnected(ISensor sensor)
{
    foreach (var baseStation in ParkingTopology.BaseStations)
    {
        if (baseStation.Sensors.Contains(sensor))
            return true;
    }
    return false;
};
```

- `CheckCollisions` – метод перевіряє чи можливий зв'язок між двома вузлами на парковці, має 2 вхідних параметра: екземпляр класу, що реалізовує інтерфейс `ISensor` і екземпляр класу, що реалізовує інтерфейс `IWirelessStation`. Працює наступним чином: перевіряє чи знаходяться об'єкти в межах досяжності один для одного а потім перевіряє чи не заважають місцеві об'єкти зв'язку:

```
bool CheckCollisions(ISensor sensor, IWirelessStation station)
{
    Line2D line = new Line2D(sensor.Position, station.Position);
    if (line.Length > Math.Min(sensor.ConnectivityRange,
        station.ConnectivityRange))
        return false;
    return ParkingSchema.CheckCollisions(line);
}
```

Клас `ParkingSchema` відповідає за схему парковки. Цей клас містить в собі геометричну схему парковки(екземпляр класу `ParkingLayout`) і описує об'єкти, що містяться на парковці(список з екземплярів класу `ParkingObject`). Клас реалізовує один метод

- `CheckCollisions` – метод перевіряє чи можливий зв'язок між двома вузлами на парковці, має один вхідний параметр: відрізок між вузлами(екземпляр класу `Line2D`). Перевіряє чи на відрізку між вузлами відсутні перешкоди, спочатку перевіряється схема парковки, а потім проводиться прохід по кожному

об'єкту на парковці і виконується перевірка чи заважає цей об'єкт зв'язку чи ні:

```
public bool CheckCollisions(Line2D line)
{
    if (!Layout.IsVisable(line))
        return false;
    foreach(var parkObj in ParkingObjects)
    {
        if(!parkObj.IsVisable(line))
            {return false;}
    }
    return true;
}
```

Клас ParkingLayout відповідає за геометричну розмітку парковки. Цей клас містить в собі список точок які задають площину парковки(список екземплярів класу Point). Клас реалізовує 4 методи:

- CreateLayout – метод відповідає за створення графічного відображення для схеми парковки.
- IsPointInside – метод перевіряє чи знаходиться точка всередині парковки, має один вхідний параметр екземпляр класу Point (двовимірна точка)

```
public bool IsPointInside(Point point)
{
    bool isInside = false;
    for (int i = 0, j = Vertexes.Count - 1; i < Vertexes.Count; j = i++)
    {
        if (((Vertexes[i].Y > point.Y) != (Vertexes[j].Y > point.Y)) &&
            (point.X < (Vertexes[j].X - Vertexes[i].X) * (point.Y -
                Vertexes[i].Y) / (Vertexes[j].Y -
                Vertexes[i].Y)+Vertexes[i].X))
        {
            isInside = !isInside;
        }
    }
    return isInside;
}
```

- configureShape – метод відповідає за параметри відображення.

- **IsVisible** – метод перевіряє чи можливий зв'язок між двома вузлами на парковці, має один вхідний параметр: відрізок між вузлами(екземпляр класу **Line2D**). Перевіряє чи на відрізку між вузлами відсутні перешкоди:

```
public bool IsVisible(Line2D connLine)
{
    for (int i = 0; i < Vertexes.Count; i++)
    {
        Point start = i == 0 ? Vertexes.Last() : Vertexes[i - 1];
        Point end = Vertexes[i];

        Line2D line = new Line2D(start, end);

        if (line.IsLineCross(connLine))
            return false;
    }

    return true;
}
```

Клас **ParkingObject** відповідає за опис об'єктів на парковці. Цей клас містить в собі список точок які задають полігон об'єкта(список екземплярів класу **Point**), вигляд форми об'єкта(з перерахування **ObjectForms**), проникність об'єкта(з перерахування **Permability**). Клас реалізовує 3 методи:

- **CreateShape**– метод відповідає за відображення.
- **IsVisible** – метод перевіряє чи можливий зв'язок між двома вузлами на парковці, має один вхідний параметр: відрізок між вузлами(екземпляр класу **Line2D**). Перевіряє чи на відрізку між вузлами відсутні перешкоди:

```
public bool IsVisible(Line2D connLine)
{
    if (Permability == Permability.Full)
        return true;
    for (int i = 0; i < Vertexes.Count; i++)
    {
        Point start = i == 0 ? Vertexes.Last() : Vertexes[i - 1];
        Point end = Vertexes[i];
        Line2D line = new Line2D(start, end);
        if (line.IsLineCross(connLine))
        {
            return false;
        }
    }
    return true;
}
```

- `configureShape` – метод відповідає за параметри відображення.

Клас `SmartParkingTopology` відповідає за опис топології мережі на парковці. Цей клас містить в собі список сенсорів які відповідають за визначення чи є автомобіль на парковці(список екземплярів класу, що реалізує інтерфейс `ISensor`), список Bluetooth-концентраторів які відповідають за отримання даних від сенсора та відправки цих даних на сервер(список екземплярів класу, що реалізує інтерфейс `IWirelessStation`).

Клас `Sensor` відповідає за опис об'єкта сенсора на парковці. Цей клас містить в собі ідентифікатор(для унікальної ініціалізації сенсора), позицію сенсора(екземпляр класу `Point`), дальність бездротового зв'язку(параметр, що використовується при обчисленні можливості підключення до концентратора), список Bluetooth-концентраторів до яких можливо підключити цей сенсор(список екземплярів класу, що реалізує інтерфейс `IWirelessStation`). Клас реалізовує один метод:

- `CreateShape`– метод відповідає за візуальне відображення сенсора.

Клас `WirelessStation` відповідає за опис об'єкта концентратора на парковці. Цей клас містить в собі ідентифікатор(для унікальної ініціалізації концентратора), позицію концентратора (екземпляр класу `Point`), дальність бездротового зв'язку(параметр, що використовується при обчисленні можливості підключення до сенсора), список сенсорів до яких можливо підключено цей концентратор(список екземплярів класу, що реалізує інтерфейс `ISensor`), кількість можливих підключень до концентратора(ціле число). Клас реалізовує 3 методи:

- `CreateShape` – метод відповідає за візуальне відображення концентратора.
- `CreateAreaShape` – метод відповідає за візуальне відображення зони покриття концентратора.
- `AddSensor`– метод відповідає за додавання сенсора до концентратора та перевірку чи можливе це підключення:

```
public bool AddSensor(ISensor sensor)
```

```

{
    if (Sensors.Count < InputConnections)
    {
        sensor.ColorBrush = ColorBrush;
        Sensors.Add(sensor);
        return true;
    }
    else
        return false;
}

```

Клас `ParkingZoneSerialization` відповідає за експорт та імпорт схеми парковки до програми, реалізує два методи:

- `SerializeParkingLayout` – метод відповідає за експорт схеми парковки в файл типу JSON.

```

SerializeParkingLayout(ParkingSchema parkingSchema, string path)
{
    string convertedNet = JsonConvert.SerializeObject(parkingSchema,
        Formatting.Indented);
    File.WriteAllText(string.Format("{0}\\{1}.json", path,
        SerializeFileName), convertedNet);
}

```

- `DeserializeParkingLayout` – метод відповідає за імпорт схеми парковки з файлу типу JSON.

```

ParkingSchema DeserializeParkingLayout(string fromFile)
{
    string JSONstring = File.ReadAllText(fromFile);
    ParkingSchema parkingSchema =
        JsonConvert.DeserializeObject<ParkingSchema>(JSONstring);
    return parkingSchema;
}

```

Клас `Line2D` відповідає за розрахунки перетинів прямих, позицій точки відносно прямої і розрахунку довжини прямої. Клас містить в собі точки початку і кінця прямої (екземпляр класу `Point`), довжину прямої та реалізує 2 функції:

- `calcLength` – метод відповідає за розрахунок довжини.

```

void calcLength()
{
    Length = Math.Sqrt(Math.Pow((_end.Y - _start.Y), 2) +
        Math.Pow((_end.X - _start.X), 2));
}

```

- `IsLineCross` – метод відповідає за перевірку перетину прямих.

```

public bool IsLineCross(Line2D line1)

```

```

{
    Line2D line2 = this;
    double x1 = line1.Start.X;
    double x2 = line1.End.X;
    double x3 = line2.Start.X;
    double x4 = line2.End.X;
    double y1 = line1.Start.Y;
    double y2 = line1.End.Y;
    double y3 = line2.Start.Y;
    double y4 = line2.End.Y;
    double eps = 0.000001;
    double del = (x1 - x2) * (y3 - y4) - (y1 - y2) * (x3 - x4);
    if (Math.Abs(del) < eps) // если паралельны
    {
        return false;
    }
    double crossX = ((x1 * y2 - y1 * x2) * (x3 - x4) - (x1 - x2)
        * (x3 * y4 - y3 * x4)) / del;
    double crossY = ((x1 * y2 - y1 * x2) * (y3 - y4) - (y1 - y2)
        * (x3 * y4 - y3 * x4)) / del;
    if((crossX >= Math.Min(x1,x2)) &&
        (crossX <= Math.Max(x1, x2))
        && (crossY >= Math.Min(y1, y2))
        && (crossY <= Math.Max(y1, y2))
        && (crossX >= Math.Min(x3, x4))
        && (crossX <= Math.Max(x3, x4))
        && (crossY >= Math.Min(y3, y4))
        && (crossY <= Math.Max(y3, y4)))
        return true;
    return false;}

```

Клас `MarkerClusterer` відповідає за розрахунок оптимального розташування комутуючих пристроїв на парковці. Працює шляхом кластеризації сенсорів в окремі кластери, а потім розраховує оптимальне розташування сенсорів базуючись на опорних точках. Клас містить в собі екземпляр класу `SmartParkingSchema` та реалізує 4 метода:

- `runClusterAlgo` – метод відповідає за початковий розподіл на кластери та запуск алгоритму знаходження оптимального розташування комутаторів на парковці:

```

void runClusterAlgo()
{
    var sensors = _smartParking.ParkingTopology.Sensors;
    List<Point> wirelessPoints = new List<Point>();
    foreach(var sensor in sensors)
    {
        wirelessPoints.Add(sensor.Position);
    }
}

```

```

for (int i = 0; i < sensors.Count; i++)
{
    for (int j = i; j < sensors.Count; j++)
    {
        Point pt = new Point((sensors[i].Position.X +
                               sensors[j].Position.X) / 2,
                               (sensors[i].Position.Y + sensors[j].Position.Y) / 2);
        wirelessPoints.Add(pt);
    }
}
ConcurrentBag<List<IWirelessStation>> stations = new
    ConcurrentBag<List<IWirelessStation>>();

Parallel.For(0, 1000, new
    ParallelOptions { MaxDegreeOfParallelism = 16 },
    (k) => makeClustering(k, wirelessPoints, stations));
List<List<IWirelessStation>> availableSolutions =
    stations.OrderBy(x
        => x.Count).ToList();
createParkingTopology(availableSolutions.FirstOrDefault());
}
}

```

- **createParkingTopology** – метод відповідає за створення топології бездротової мережі розумної парковки, перевіряє чи можливе підключення комутатора до датчика і встановлює це підключення:

```

createParkingTopology(List<IWirelessStation> availableSolutions)
{
    foreach (var stOld in availableSolutions)
    {
        Point position = new Point(stOld.Position.X, stOld.Position.Y);
        WirelessStation station = new WirelessStation(position,
            stOld.ConnectivityRange, stOld.InputConnections);
        foreach (var sensor in stOld.Sensors)
            station.AddSensor(sensor);
        _smartParking.ParkingTopology.BaseStations.Add(station);
    }
}

```

- **makeClustering** – метод відповідає за розподіл сенсорів на парковці на кластери:

```

void makeClustering(int k, List<Point> wirelessPoints,
    ConcurrentBag<List<IWirelessStation>> stationsVars)
{
    List<IWirelessStation> stations = new List<IWirelessStation>();
    var sensors = _smartParking.ParkingTopology.Sensors;
    int randCounter = 0;
    lock (_locker)
    {
        randCounter = Params.RandCounter++;
    }
}

```

```

}
Random r = new Random(randCounter);
foreach (Point pt in wirelessPoints.OrderBy(x => r.Next()))
{
    var newStation = new WirelessStation(pt);
    for(int i=0;i<_smartParking.ParkingTopology.Sensors.Count;i++)
    {
        var sensor = sensors[i];
        // iterate clusters and collect candidates
        if (!isSensorConnected(stations, sensor))
        {
            if (_smartParking.CheckCollisions(sensor, newStation))
            {
                if (!newStation.AddSensor(sensor))
                    break;
            }
        }
    }
    if (newStation.Sensors.Count != 0)
    {
        stations.Add(newStation);
    }
}
stationsVars.Add(stations);}

```

- `isSensorConnected` – метод перевіряє чи підключений сенсор до комутатора на парковці, має вхідний параметр екземпляр класу, що реалізовує інтерфейс `ISensor`. Метод проходить по всім бездротовим комутаторам в мережі та перевіряє чи містять вони підключення до цього сенсора:

```

public bool IsSensorConnected(ISensor sensor)
{
    foreach (var baseStation in ParkingTopology.BaseStations)
    {
        if (baseStation.Sensors.Contains(sensor))
            return true;
    }
    return false;
};

```

3.2 Тестування програми

Тестування проводитиметься на декількох прикладах, перший тест проведемо використовуючи дану схему парковки, що зображена на рисунку 3.3.

В першому прикладі враховуємо, що відстань між датчиками на паркомісцях дорівнює 3 метри, а відстань бездротового зв'язку становить 20 метрів, також важливим параметром є кількість можливих підключень до бездротового комутатора, яка становить 24(може змінюватись в залежності від приладу, за основу взяті середні показники). Також потрібно сказати про умовні позначення програм: світло-червоний колір з суцільним контуром позначає об'єкт, що є не проникним для сигналу; світло-синій колір з пунктирним контуром позначає об'єкт схеми парковки, що не заважає поширенню сигналу; трикутником позначені самі сенсори; трикутник з радіальною зоною покриття – це бездротовий комутатор

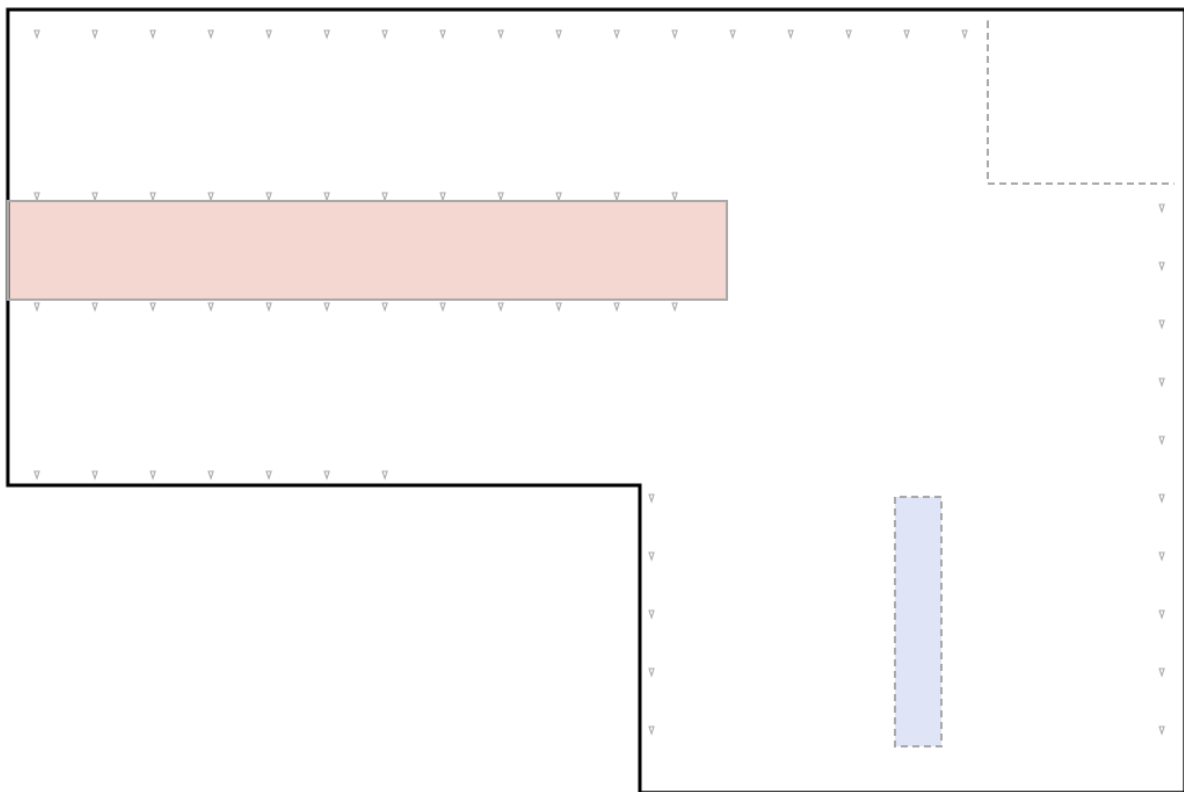


Рисунок 3.3 – Схема парковки для першого прикладу

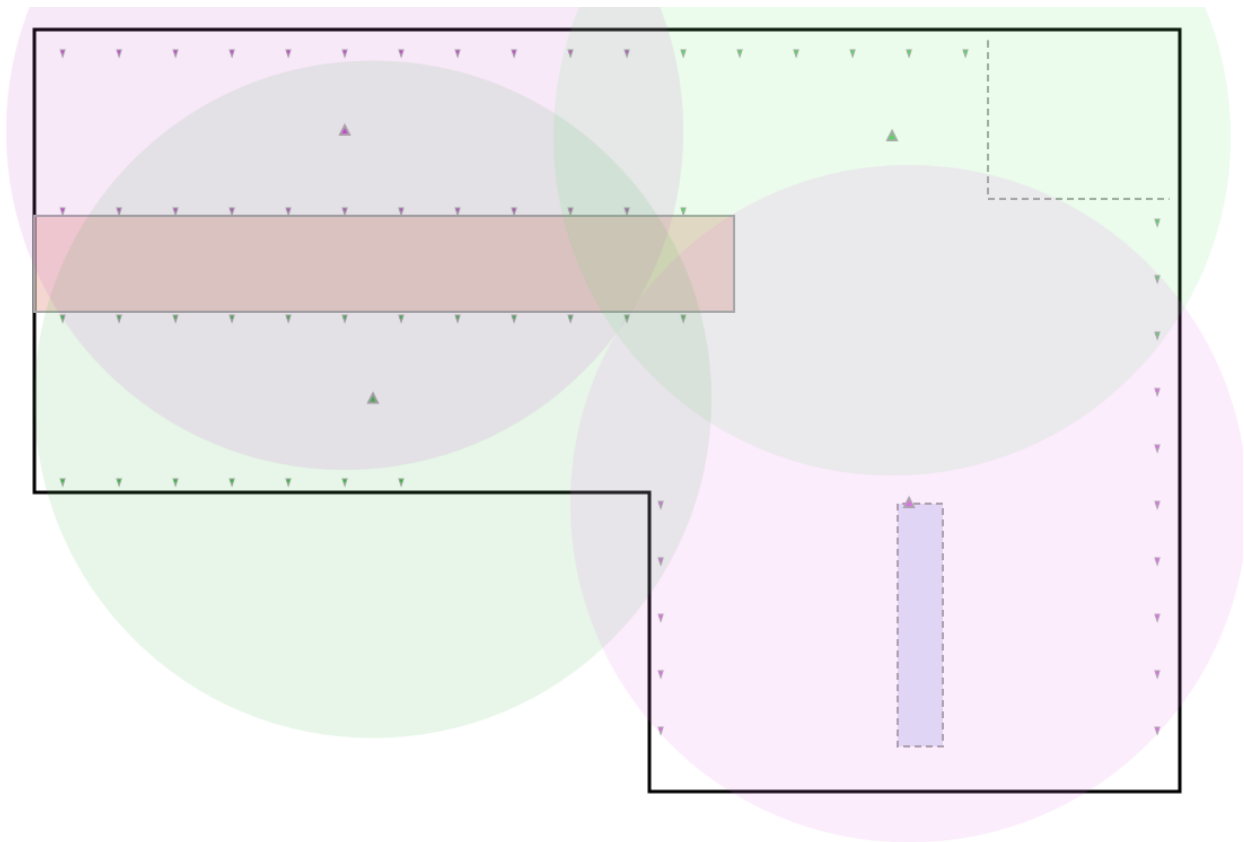


Рисунок 3.4 – Результат роботи програми

Отриманий результат зображений на рисунку 3.4. Як бачимо комутатори розташувалися доволі вигідно і неможливо видалити жодний з них лишивши всі датчики покритими. Проведемо дослідження впливу кількості ітерацій на оптимальність рішення.

Таблиця 3.1 – Залежність оптимальності рішення від кількості ітерацій

Кількість ітерацій	Час виконання (с)	Кількість комутаторів
1000	10.23	4
500	7.65	4
300	5.12	5
100	2.75	5

Як видно з табл. 3.1 для даного випадку достатньо 500 ітерацій для отримання оптимального рішення, хоча це дуже залежить від схеми парковки та від кількості паркомісць.

Для додаткового прикладу збільшимо кількість паркомісць, припустивши, що відстань між датчиками буде 1.5 метри замість 3, тобто аналог парковки для мотоциклів та інших невеликих засобів пересування. За основу схеми візьмемо ту ж саму схему парковки, результати зображено на рисунку 3.5.

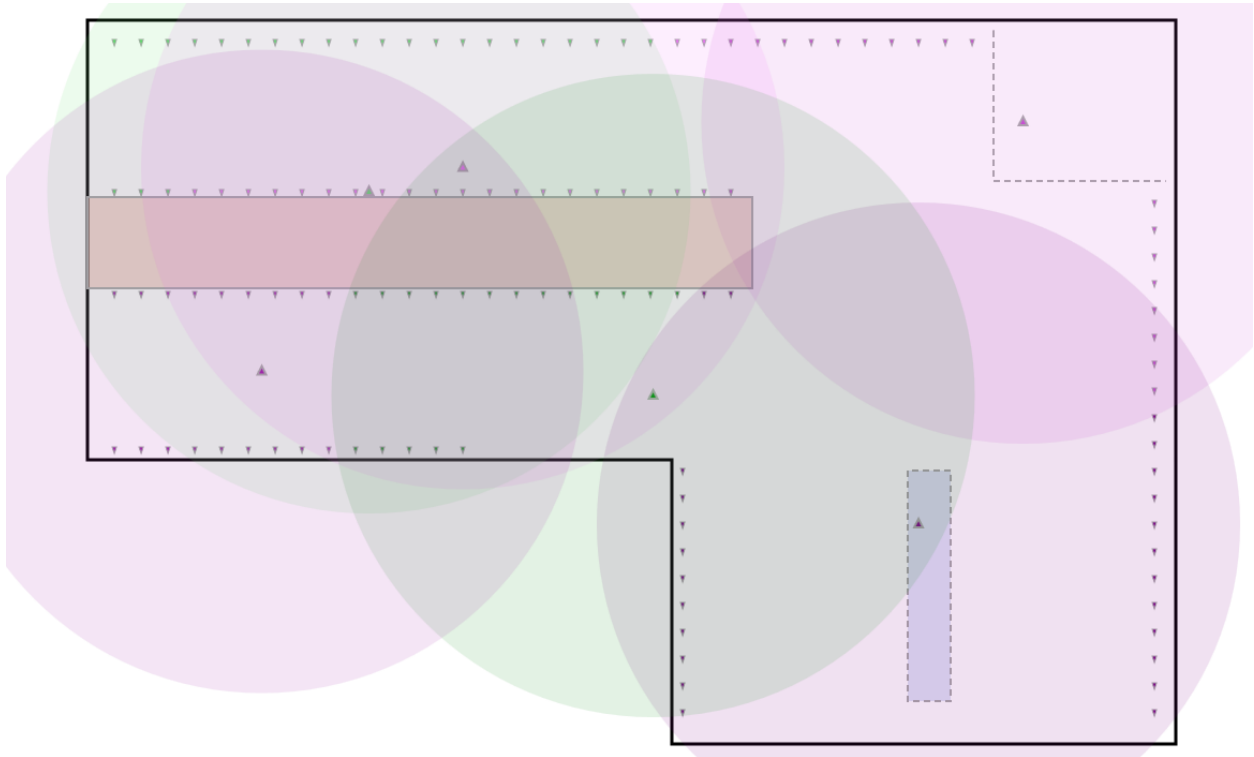


Рисунок 3.5 – Результат роботи програми при більшій щільності датчиків

На перший погляд результат здається не дуже оптимальним оскільки зони доволі сильно перекривають одна одну, проте на кожен комутатор закладене обмеження в 24 вхідних підключення, а кількість паркомісць на парковці дорівнює 125, отже щоб покрити 125 сенсорів необхідно мати таку кількість комутаторів: $125/24 = 5.2$, округлюється до більшого, тобто до 6. Отримане рішення має 6 комутаторів, тобто рішення є повністю оптимальним.

Другий тест проведемо використовуючи дану схему парковки, що зображена на рисунку 3.6.

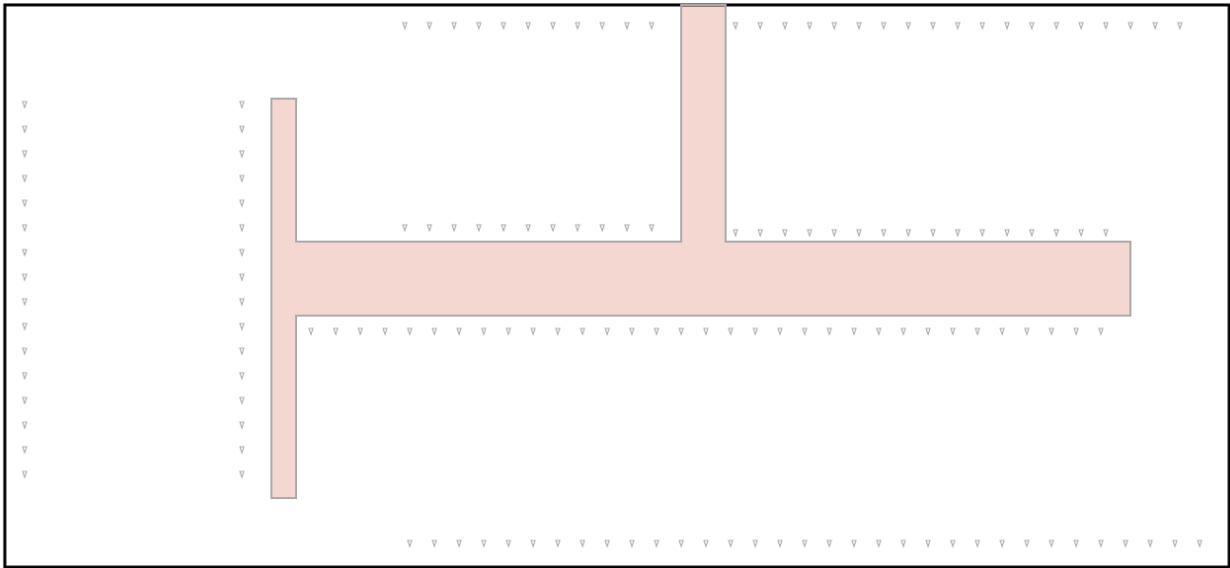


Рисунок 3.6 – Схема парковки для нового прикладу

В другому прикладі враховуємо, що відстань між датчиками на парко місцях дорівнює 1.5 метри, а відстань бездротового зв'язку становить 20 метрів, також важливим параметром є кількість можливих підключень до бездротового комутатора, яка становить 24(може змінюватись в залежності від приладу, за основу взяті середні показники). Результат виконання програми зображено на рисунку 3.7.

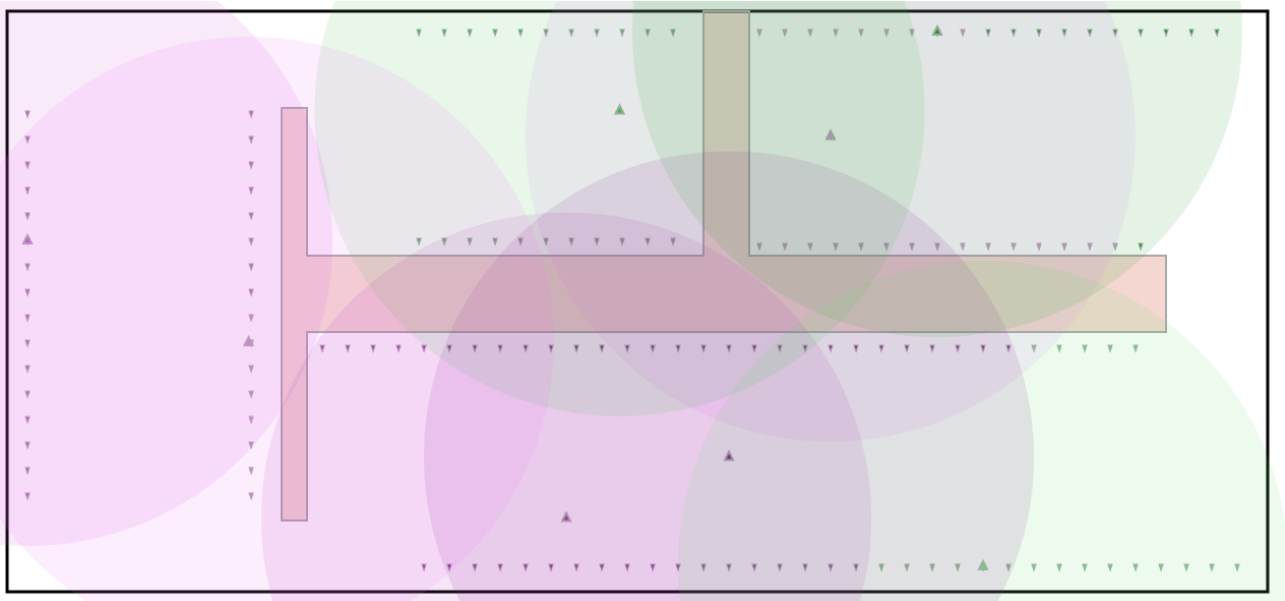


Рисунок 3.7 – Результат роботи програми при відстані 3 метри між датчиками

В даному випадку програмі знадобилось 8 комутаторів для того щоб забезпечити покриттям усі датчики на даній парковці. Парковка містить 155 парко місць, які потребують мінімум 7 комутаторів (один комутатор може підключити 24 сенсора). Хоч 7 це мінімум комутаторів для даної кількості, проте складна схема парковки не дозволила отримати таке число, також візуально не має очевидного методу як зменшити кількість комутаторів, тому можна вважати, що програма впоралася зі своїм завданням.

Сформуємо таблицю з результатами виконання для цього прикладу з різною щільністю парко місць, а саме 3 і 1.5 метрів на парко місце(ширина парко місця), результати відображені в табл. 3.2. Також на рисунку 3.8 зображено результат виконання програми для відстані в 3 метри між датчиками.

Таблиця 3.2 – Залежність кількості комутаторів від кількості місць

Ширина парко місця(м)	Кількість Парко місць	Час виконання(с)	К-сть комутаторів
1.5	155	36.95	8
3	80	23.12	5

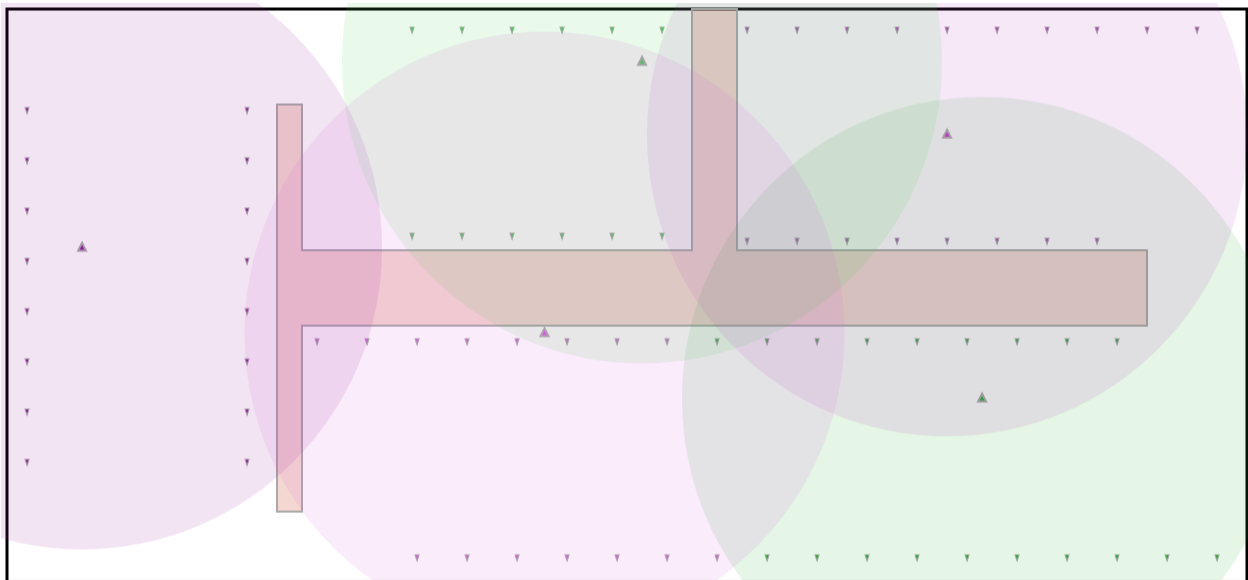


Рисунок 3.8 – Результат роботи програми при відстані 1.5 метрів між датчиками

3.3 Висновки

В цьому розділі було наведено результати проведеного дослідження. Було наведено зображення з результатами дослідження і сформовано декілька таблиць де проаналізовано вплив кількості датчиків на швидкість роботи програми, а також вплив кількості ітерацій на оптимальність рішення.

Також описано розроблену програму, що складається з двох проектів, один це динамічно приєднувана бібліотека з основними обчисленнями, а інший слугує для відображення результатів обрахунків.

Розроблена програма надає більш широкі можливості для обрахунку оптимального позиціонування датчиків на автостоянці ніж існуюче рішення, що використовувалося при розгортанні прототипу системи Smart Parking в Алжирі, адже програма дозволяє завантажувати довільну схему парковки, налаштовувати параметри мережевих пристроїв та враховує перешкоди для сигналу на парковці. Програма, що використовувалась при розгортанні прототипу в Алжирі основним недоліком мала відсутність можливості змінити схему парковки та вона не враховувала перешкоди для сигналу на самій парковці.

4 РОЗРОБЛЕННЯ СТАРТАП-ПРОЕКТУ “SMART PARKING COVERAGE”

Стартап як форма малого ризикового (венчурного) підприємництва впродовж останнього десятиліття набула широкого розповсюдження у світі через зниження бар'єрів входу в ринок (із появою Інтернету як інструменту комунікацій та збуту стало простіше знаходити споживачів та інвесторів, займатись пошуком ресурсів, перетинати кордони між ринками різних країн), і вважається однією із наріжних складових інноваційної економіки, оскільки за рахунок мобільності, гнучкості та великої кількості стартап-проектів загальна маса інноваційних ідей зростає.

Проте створення та ринкове впровадження стартап-проектів відзначається підвищеною мірою ризику, ринково успішними стає лише невелика частка, що за різними оцінками складає від 10% до 20%. Ідея стартап-проекту, взята окремо, не вартує майже нічого: головним завданням керівника проекту на початковому етапі його існування є перетворення ідеї проекту у працюючу бізнес-модель, що починається із формування концепції товару (послуги) для визначеної клієнтської групи за наявних ринкових умов. Розроблення та виведення стартап-проекту на ринок передбачає здійснення низки кроків, в межах яких визначають ринкові перспективи проекту, графік та принципи організації виробництва, фінансовий аналіз та аналіз ризиків і заходи з просування пропозиції для інвесторів.

4.1 Опис ідеї проекту

В межах підпункту було проаналізовано і подано у вигляді таблиць:

- зміст ідеї (що пропонується);
- можливі напрямки застосування;
- основні вигоди, що може отримати користувач товару (за кожним напрямком застосування);
- чим відрізняється від існуючих аналогів та замінників;

Таблиця 4.1 Опис ідеї стартап-проекту

<i>Зміст ідеї</i>	<i>Напрямки застосування</i>	<i>Вигоди для користувача</i>
Ідея полягає в тому, щоб створити застосунок, за допомогою якого можна було б вирішити проблему оптимального розташування датчиків в системі Smart Parking. Застосунок має надавати користувачу можливість завантажувати схему парковки за допомогою XML чи Json файлу	1. Використання для економії коштів на розгортання бездротової мережі для системи Smart Parking.	Зменшує вартість розгортання бездротової сенсорної мережі шляхом оптимізації кількості комутуючих пристроїв на автостоянці.
	2. Використання для автоматичного розрахунку топології мережі.	При проектуванні бездротової мережі знімає з інженерів задачу пошуку оптимального розташування комутуючих пристроїв. Дозволяє враховувати особливості схеми автомобільної парковки такі як стіни, колони і інші об'єкти що можуть завадити поширенню сигналу.

У таблиці 4.1 було описано ідею стартапу «Smart Parking Coverage ». Проект може бути використаний як інструмент для пошуку оптимального розташування комутуючих пристроїв на автостоянці. Цей проект дозволяє суттєво знизити вартість на обладнання та спростити побудову мережі для інженерів, так як надає можливість враховувати особливості схеми автостоянки.

Таблиця 4.2 Визначення сильних, слабких та нейтральних характеристик ідеї проекту

№ п/ п	Техніко- економічні характерис- тики ідеї	(потенційні) товари/концепції конкурентів				W (слабка сторон а)	N (нейт- ральна сторон а)	S (сильна сторон а)
		Мій проект	Конку- рент 1	Конку- рент 2	Конку- рент 3			
1.	Форма виконання	Засто- сунок	Засто- сунок	Веб- додаток	Засто- сунок		+	
2.	Собівартість	Низька	Висока	Низька	Висока			+
3.	Кросплат- формність	Ні	Ні	Так	Ні	+		
4.	Наявність зручного інтерфейсу користувача	Так	Так	Так	Так		+	
5.	Можливість завантаження схеми з XML чи JSON файлу	Так	Ні	Ні	Так			+
6.	Можливість отримати рішення в форматі XML і JSON для обробки в іншому ПО	Так	Ні	Ні	Ні			+
7.	Можливість змінювати параметри мережевих пристроїв	Так	Так	Ні	Так		+	
8.	Портативність	Так	Ні	Так	Ні			+

Сильними сторонами проекту є низька собівартість, можливість завантаження схеми з XML чи JSON файлу, можливість отримати рішення в форматі XML і JSON для обробки в іншому ПО, портативність. Слабкою стороною є відсутність кросплатформності, хоча таку можливість надає лише один з конкурентів, нейтральними – форма виконання, зручний інтерфейс користувача і можливість редагувати параметри мережевих пристроїв. Отож, система є конкурентоспроможною.

4.2 Технологічний аудит ідеї проекту.

В межах даного підрозділу необхідно провести аудит технології, за допомогою якої можна реалізувати ідею проекту (технології створення товару). Основними критеріями вибору технологій були: підтримка парадигми ООП, безкоштовне середовище розробки.

Таблиця 4.3 Технологічна здійсненність ідеї проекту

<i>№ n/n</i>	<i>Ідея проекту</i>	<i>Технології реалізації</i>	<i>Наявність технологій</i>	<i>Доступність технологій</i>
1.	Створення бібліотеки для обрахунків	C#, .Net	Наявна	Безкоштовна, доступна
2.	Створення інтерфейсу користувача для зручної взаємодії	WPF, XAML	Наявна	Безкоштовна, доступна

Обрані технології реалізації ідеї проекту: C#, .Net через повну безкоштовність фреймворку та наявність докладної документації, наявність досвіду роботи розробників з даною технологією; WPF, XAML через простоту використання, безкоштовність та зручність в створенні інтерфейсу користувача.

4.3 Аналіз ринкових можливостей запуску стартап-проекту.

Визначення ринкових можливостей, які можна використати під час ринкового впровадження проекту, та ринкових загроз, які можуть перешкодити реалізації проекту, дозволяє спланувати напрями розвитку проекту із урахуванням стану ринкового середовища, потреб потенційних клієнтів та пропозицій проектів-конкурентів.

Спочатку було проведено аналіз попиту: наявність попиту, обсяг, динаміка розвитку ринку (табл. 4.4).

Таблиця 4.4 Попередня характеристика потенційного ринку стартап-проекту

№ п/п	Показники стану ринку (найменування)	Характеристика
1.	Кількість головних гравців, од	4
2.	Загальний обсяг продаж, грн/ум.од	18000 грн./ум.од
3.	Динаміка ринку (якісна оцінка)	Зростає
4.	Наявність обмежень для входу (вказати характер обмежень)	Немає
5.	Специфічні вимоги до стандартизації та сертифікації	Немає
6.	Середня норма рентабельності в галузі (або по ринку), %	$R = (3000000 * 100) / (1000000 * 12) = 25\%$

Отже, було проаналізовано наявність попиту, обсяг, динаміку розвитку ринку. Обмеження для входу на ринок відсутні, динаміка ринку зростає, галузь є рентабельною.

Надалі визначаються потенційні групи клієнтів, їх характеристики, та формується орієнтовний перелік вимог до товару для кожної групи (табл. 4.5).

Таблиця 4.5 Характеристика потенційних клієнтів стартап-проекту

<i>№ n/n</i>	<i>Потреба, що формує ринок</i>	<i>Цільова аудиторія (цільові сегменти ринку)</i>	<i>Відмінності у поведінці різних потенційних цільових груп клієнтів</i>	<i>Вимоги споживачів до товару</i>
1.	Необхідне програмне забезпечення, що надає можливість розрахувати оптимальну топологію мережі на парковці.	Потенційними цільовими групами є державні служби, приватні підприємства, інші організації, що надають послуги паркінгу, специфіка роботи яких потребує моніторингу та планування місць на парковці.	Цільова група мережеві інженери на підприємствах, відмінностей між групами не має, всі можуть використовувати застосунок	Рішення повинне бути здатним враховувати особливості місцевості на парковці(перешкоди, стіни, колони), бути здатним імпортувати та експортувати схему парковки, мати можливість налаштування характеристик пристроїв для обчислень.

Згідно проведеної характеристики потенційних клієнтів стартап-проекту впливає, що на ринку є затребуваним програмне забезпечення(застосунок) для розрахунку оптимальної топології мережі і потенційними цільовими групами є державні служби, приватні підприємства, інші організації, що надають послуги паркінгу, специфіка роботи яких потребує моніторингу та планування місць на парковці.

Після визначення потенційних груп клієнтів проводиться аналіз ринкового середовища: складаються таблиці факторів, що сприяють ринковому впровадженню проекту, та факторів, що йому перешкоджають (табл. 4.6-4.7). Фактори в таблиці подавати в порядку зменшення значущості.

Таблиця 4.6 Фактори загроз

<i>№ n/n</i>	<i>Фактор</i>	<i>Зміст загрози</i>	<i>Можлива реакція компанії</i>
1.	Конкуренція	Вихід на ринок великої компанії	1. Вихід з ринку 2. Запропонувати великій компанії поглинути себе 3. Передбачити додаткові переваги власного застосування для того, щоб повідомити про них саме після виходу міжнародної компанії на ринок
2.	Зміна потреб користувачів	Користувачам необхідне програмне забезпечення з іншим функціоналом	1. Передбачити можливість додавання нового функціоналу до створюваного ПЗ
3.	Створення нових технічних засобів	Створення нових приладів чи мережевого обладнання, що не буде потребувати оптимізації розташування	1. Модернізація алгоритму проекту, перехід до нової парадигми
4.	Надходження на ринок альтернативних продуктів	Перехід користувачів нашого товару на інший продукт	Впровадження нового функціоналу, якого немає у конкурентів
5.	Уповільнення росту ринку	Скорочення користувачів продуктів, що тільки виходять на ринок	Інвестиції у впровадження ефективної реклами продукту

Отже, було проаналізовано фактори загроз ринкового впровадження проекту, серед яких: конкуренція, уповільнення росту ринку, зміна потреб користувачів, створення нових технічних засобів та приладів для організації мережі та надходження на ринок альтернативних продуктів. Було також запропоновано можливі реакції компанії.

Таблиця 4.7 Фактори можливостей

<i>№ n/n</i>	<i>Фактор</i>	<i>Зміст можливості</i>	<i>Можлива реакція компанії</i>
1.	Стрімкий ріст попиту на інструменти для розгортання системи Smart Parking	Наявність попиту на інструменти для організації і розрахунку необхідних параметрів при розгортанні системи Smart Parking	Змога запропонувати продукт більшої кількості потенційних користувачів
2.	Поява нових технологій	Надання нового функціоналу для розрахунку оптимальної топології мережі враховуючи нові технології	Розробка нового функціоналу у вигляді нової бібліотеки для обрахунків під нову специфікацію мережі
3.	Стрімке зростання росту ринку	Компаніям, що тільки виходять на ринок, буде простіше отримати клієнтів	Змога запропонувати продукт більшої кількості потенційних користувачів
4.	Інтеграція існуючими популярними додатками	Можливість використати клієнтську базу більш відомого додатку	Інтеграція системи Smart Parking з сервісами від Google, таким я Waze, чи схожими сервісами – 2Gis
5.	Розширення асортименту можливих послуг	Поява нового функціоналу, що привабить нових користувачів	Розробка нового функціоналу, що є потребою певної групи користувачів

Отже було наведено фактори можливостей ринкового впровадження проекту (табл. 4.7), серед яких: стрімкий ріст попиту на інструменти для розгортання системи Smart Parking, поява нових технологій, стрімке зростання росту ринку, інтеграція з існуючими популярними додатками, розширення асортименту можливих послуг; було також запропоновано можливі реакції компанії.

Надалі було проведено аналіз пропозиції: визначили загальні риси конкуренції на ринку (табл. 4.8).

Таблиця 4.8 Ступеневий аналіз конкуренції на ринку

<i>Особливості конкурентного середовища</i>	<i>В чому проявляється дана характеристика</i>	<i>Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)</i>
1. Вказати тип конкуренції - нецінова конкуренція.	Існує декілька конкурентів, які повторюють деякі функції нашої системи	Підтримка якості застосунка та постійні нововведення
2. За рівнем конкурентної боротьби - міжнародний	Фірми-конкуренти знаходяться як в нашій країні так і в інших країнах.	Додати можливість вибору мови застосунку, щоб легше було у майбутньому вийти на міжнародний ринок
3. За галузевою ознакою - внутрішньогалузева	Застосунок використовується лише всередині даної галузі	Постійне вдосконалення застосунку в залежності від новинок в мережевих технологіях
4. Конкуренція за видами товарів: - товарно-родова	Застосунки конкурентів виконують подібні функції але досить відрізняються від нашої	Створити продукт, врахувавши сильні і слабкі сторони конкурентів.
5. За характером конкурентних переваг - нецінова	Вдосконалення технології створення застосунку, щоб собівартість була нижчою	Використання менш дорогих технологій для розробки, ніж використовують конкуренти
6. За інтенсивністю - не марочна	Бренди відсутні	-

Проведено ступеневий аналіз конкуренції на ринку (табл. 4.8), де було визначено особливості конкурентного середовища та їх вплив а діяльність підприємства. Однією з найбільш важливих дій компанії для досягнення конкурентоспроможності є необхідність створити основу застосунку таким чином,

щоб можна було легко переробити його для обрахунку з врахуванням новітніх мережевих технологій.

Після аналізу конкуренції проводиться більш детальний аналіз умов конкуренції в галузі (табл. 9).

Таблиця 4.9 – Аналіз конкуренції в галузі за М. Портером

<i>Складові аналізу</i>	<i>Прямі конкуренти в галузі</i>	<i>Потенційні конкуренти</i>	<i>Постачальники</i>	<i>Клієнти</i>	<i>Товари-замінники</i>
	<i>Навести перелік прямих конкурентів</i>	<i>Визначити бар'єри входження в ринок</i>	<i>Визначити фактори сили постачальників</i>	<i>Визначити фактори сили споживачів</i>	<i>Фактори загроз з боку замінників</i>
Висновки :	Існує 3 конкуренти на ринку. Найбільш схожим за виконанням є конкурент 1, так як його рішення також представлене у вигляді застосунку і дозволяє змінювати характеристик приладів, відповідно до вимог мережі	Так, можливості для входу на ринок є, бо наше рішення поєднує в собі велику кількість можливостей, має зручний інтерфейс, а також надає можливості для експорту та імпорту схеми парковки	Постачальник и відсутні.	Важливим для користувача є зручність у користуванні та адекватність результату	Товари-замінники можуть використати більш дешеву технологію створення застосунку та зменшити собівартість товару

Було здійснено аналіз конкуренції в галузі за М. Портером, в результаті чого було визначено, що існує 3 конкуренти на ринку. Найбільш схожим за виконанням є конкурент 1, так як його рішення також представлене у вигляді застосунку та дозволяє змінювати характеристики приладів, відповідно до вимог мережі, але

можливості для входу на ринок є, бо наше рішення має можливість експорту та імпорту схеми парковки в форматах XML та JSON.

За результатами аналізу таблиці робиться висновок щодо принципової можливості роботи на ринку з огляду на конкурентну ситуацію. Також робиться висновок щодо характеристик (сильних сторін), які повинен мати проект, щоб бути конкурентоспроможним на ринку. Другий висновок враховується при формулюванні переліку факторів конкурентоспроможності) На основі аналізу конкуренції, проведеного в табл. 9, а та- кож із урахуванням характеристик ідеї проекту (табл. 4.2), вимог споживачів до товару (табл. 5) та факторів маркетингового середовища (табл. 4.6, табл. 4.7) визначається та обґрунтовується перелік факторів конкурентоспроможності. Аналіз оформлюється за табл. 10.

Таблиця 4.10 - Обґрунтування факторів конкурентоспроможності

<i>№ п/п</i>	<i>Фактор конкурентоспроможності</i>	<i>Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проектів значущим)</i>
1.	Можливість налаштовувати під параметри мережевих пристроїв	Дозволяє користувачам здійснювати гнучке налаштування застосунку для використання в специфічних мережах
2.	Можливість імпортувати та експортувати схему парковки з топологією мережі	Дозволяє створювати схему парковки в певному графічному редакторі і потім зручно імпортувати в наш застосунок
3.	Портативність	Дозволяє користуватися застосунком без попереднього встановлення.
4.	Простота інтерфейсу користувача	Інтуїтивно зрозумілий інтерфейс з простим доступом до найважливіших функцій даного сервісу.

У табл. 4.10 наведено обґрунтування факторів конкурентоспроможності, серед яких: можливість налаштовувати під параметри мережевих пристроїв, можливість імпортувати та експортувати схему парковки з топологією мережі,

портативність та простота інтерфейсу користувача. Було також наведено обґрунтування цих факторів.

За визначеними факторами конкурентоспроможності (табл. 4.10) проводиться аналіз сильних та слабких сторін стартап-проекту (табл. 4.11).

У наступній таблиці наведено проведення аналізу сильних та слабких сторін стартап-проекту, факторами конкурентоспроможності виступили такі: можливість налаштовувати під параметри мережевих пристроїв, можливість імпортувати та експортувати схему парковки з топологією мережі, портативність та простота інтерфейсу користувача.

Таблиця 4.11 - Порівняльний аналіз сильних та слабких сторін проекту

№ п/п	Фактор конкурентоспроможності	Бали 1-20	Рейтинг товарів-конкурентів у порівнянні з нашим підприємством						
			-3	-2	-1	0	1	2	3
1.	Можливість налаштовувати під параметри мережевих пристроїв	20			+				
2.	Можливість імпортувати та експортувати схему парковки з топологією мережі	20		+					
3.	Портативність	15		+					
4.	Простота інтерфейсу користувача	15				+			

Отже, серед сильних сторін проекту можна виділити наступні: можливість налаштовувати під параметри мережевих пристроїв, можливість імпортувати та експортувати схему парковки з топологією мережі, портативність. Серед нейтральних сторін можна виділити простоту інтерфейсу користувача.

Перелік ринкових загроз та ринкових можливостей складається на основі аналізу факторів загроз та факторів можливостей маркетингового середовища. Ринкові загрози та ринкові можливості є наслідками (прогнозованими результатами) впливу факторів, і, на відміну від них, ще не є реалізованими на ринку та мають певну ймовірність здійснення. Наприклад: зниження доходів потенційних споживачів – фактор загрози, на основі якого можна зробити прогноз щодо посилення значущості цінового фактору при виборі товару та відповідно, – цінової конкуренції (а це вже – ринкова загроза).

У наступній таблиці буде проілюстровано SWOT-аналіз стартап-проекту, тобто його слабкі та сильні сторони, можливості та загрози виходу на ринок.

Таблиця 4.12 – SWOT-аналіз стартап-проекту

Сильні сторони: можливість налаштовувати під параметри мережевих пристроїв, можливість імпортувати та експортувати схему парковки з топологією мережі, портативність	Слабкі сторони: висока складність реалізації проекту через необхідність постійного доопрацювання під найсучасніші мережеві технології
Можливості: стрімкий ріст попиту на інструменти для розгортання системи Smart Parking, поява нових технологій, інтеграція з існуючими популярними додатками, стрімке зростання росту ринку	Загрози: створення нових технічних засобів, конкуренція, зміна потреб користувачів, надходження на ринок альтернативних продуктів, уповільнення росту ринку

На основі SWOT-аналізу розробляються альтернативи ринкової поведінки (перелік заходів) для виведення стартап-проекту на ринок та орієнтовний оптимальний час їх ринкової реалізації з огляду на потенційні проекти конкурентів, що можуть бути виведені на ринок. Визначені альтернативи аналізуються з точки зору строків та ймовірності отримання ресурсів.

В табл. 4.13 наведено альтернативи ринкового впровадження стартап-проекту.

Таблиця 4.13 – Альтернативи ринкового впровадження стартап-проекту

№ п/п	Альтернатива (орієнтовний комплекс заходів) ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
1.	Створення кросплатформного додатку за допомогою технологій Xamarin та веб додатку використовуючи ASP Net	90%	8 місяців
2.	Створення розповсюджуваної бібліотеки використовуючи .Net	35%	6 місяців

З означених альтернатив обирається та, для якої: а) отримання ресурсів є більш простим та ймовірним; б) строки реалізації – більш стислими. Тому обираємо альтернативу (створення кросплатформного додатку за допомогою технологій Xamarin та веб додатку використовуючи ASP Net).

4.4 Розробка ринкової стратегії проекту

Розроблення ринкової стратегії першим кроком передбачає визначення стратегії охоплення ринку: опис цільових груп потенційних споживачів.

Таблиця 4.14 - Вибір цільових груп потенційних споживачів

№ п/п	Опис профілю цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в межах цільової групи (сегменту)	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
1.	Торгові центри	Спрощення обрахунку оптимальної топології мережі	Великий	Існує 3 конкуренти, які надають схожі, але дорожчі та менш оптимізовані рішення.	Швидкодія, зручний користувачький інтерфейс, можливість експорту та імпорту схеми парковки
2.	Підприємства	Спрощення обрахунку оптимальної топології мережі	Великий		Швидкодія, зручний користувачький інтерфейс, можливість експорту та імпорту схеми парковки
Які цільові групи обрано: обираємо підприємства та торгові центри					

За результатами аналізу потенційних груп споживачів (сегментів) автори ідеї обирають цільові групи, для яких вони пропонуватимуть свій товар, та визначають стратегію охоплення ринку. Для роботи в обраних сегментах ринку необхідно сформулювати базову стратегію розвитку. За М. Портером, існують три базові

стратегії розвитку, що відрізняються за ступенем охоплення цільового ринку та типом конкурентної переваги, що має бути реалізована на ринку (за витратами або визначними якостями товару).

В наступній таблиці буде описано базові стратегії розвитку:

Таблиця 4.15 – Визначення базової стратегії розвитку

<i>№ п/ п</i>	<i>Обрана альтернатива розвитку проекту</i>	<i>Стратегія охоплення ринку</i>	<i>Ключові конкурентоспро- можні позиції відповідно до обраної альтернативи</i>	<i>Базова стратегія розвитку</i>
1.	Створення кросплатформного додатку за допомогою технологій Xamarin та веб сервісу використовуючи ASP Net	Ринкове позиціонування	Швидкодія, простота у користуванні, реалізація у двох варіантах (веб-сервіс та додаток). Можливість інтеграції в інші продукти завдяки веб сервісу що підтримує REST API.	Диференціація

Було обрано таку альтернативу розвитку проекту: створення кросплатформного додатку за допомогою технологій Xamarin та веб сервісу використовуючи ASP Net, адже завдяки цим технологіям можна досягнути ключових конкурентоспроможних позицій кінцевого продукту.

В табл. 4.16 буде визначено базову стратегію конкурентної поведінки.

Таблиця 4.16 - Визначення базової стратегії конкурентної поведінки

<i>№ п/п</i>	<i>Чи є проект «першопрохідцем» на ринку?</i>	<i>Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?</i>	<i>Чи буде компанія копіювати основні характеристики товару конкурента, і які?</i>	<i>Стратегія конкурентної поведінки</i>
1.	Ні	Так	Буде, а саме: 1) наявність зручного інтерфейсу користувача (конкурент 1, 2, 3) 2) можливість змінювати характеристики мережевих пристроїв (конкурент 1, 3)	Займання конкурентної ніші

Отже, було визначено базову стратегію конкурентної поведінки як зайняття конкурентної ніші. Наш проект не є «першопрохідцем», але є всі передумови для заняття своєї ніші на ринку. Даний проект буде намагатись перетягнути частину користувачів від конкурентів. Також ми плануємо взяти найбільш вдалі рішення від проектів конкурентів з подальшим їх вдосконаленням.

Визначимо стратегію позиціонування у табл. 4.17, що полягає у формуванні ринкової позиції (комплексу асоціацій), за яким споживачі мають ідентифікувати торгівельну марку/проект.

Таблиця 4.17 - Визначення стратегії позиціонування

<i>№ п/п</i>	<i>Вимоги до товару цільової аудиторії</i>	<i>Базова стратегія розвитку</i>	<i>Ключові конкурентоспроможні позиції власного стартап- проекту</i>	<i>Вибір асоціацій, які мають сформувавши комплексну позицію власного проекту (три ключових)</i>
1.	Швидкодія, простота у користуванні, можливість налаштувати параметри мережевих пристроїв, можливість експортувати чи імпортувати схему парковки	Диференціація	Швидкодія, простота у користуванні, можливість налаштувати параметри мережевих пристроїв, можливість експортувати чи імпортувати схему парковки, портативність дозволить зручно користуватися застосунком та вирішувати необхідну задачу	Швидкодія, простота, якість отриманого рішення

В таблиці вище описано стратегію позиціонування даного стартап-проекту. Описано основні вимоги цільової аудиторії до товару, а саме простоту у користуванні, можливість налаштувати параметри мережевих пристроїв, якість отриманого рішення. Визначено базову стратегію розвитку (диференціація). Сформовано перелік ключових позицій конкурентоспроможності даного проекту.

4.5 Розробка маркетингової програми

Першим кроком є формування маркетингової концепції товару, який отримає споживач. Для цього у табл. 4.18 потрібно підсумувати результати попереднього аналізу конкурентоспроможності товару.

Таблиця 4.18 - Визначення ключових переваг концепції потенційного товару

<i>№ n/n</i>	<i>Потреба</i>	<i>Вигода, яку пропонує товар</i>	<i>Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити)</i>
1.	Наявність зручного інтерфейсу користувача	Додаток реалізований у вигляді WPF застосунку, що обраховує оптимальне розташування сенсорів на парковці	Перевага в зручності інтерфейсу та можливості додавання нових модулів до застосунка.
2.	Можливість завантаження схеми парковки	Можливість завантаження схеми парковки в застосунок за допомогою XML чи JSON файлу в правильному форматі	Користувачі мають змогу працювати розробити схему парковки в спеціальному графічному редакторі та завантажити її до застосунка
3.	Можливість експорту отриманого рішення	Можливість експорту отриманого рішення в форматі XML чи JSON	Користувачі мають змогу працювати з запропонованим рішенням в спеціальному графічному редакторі і вносити свої певні корективи
4.	Можливість змінювати параметри мережевих пристроїв	Адаптивність до різних мережевих пристроїв	Користувачі мають змогу змінювати параметри мережевих пристроїв

Отже бачимо, що проект має ключові переваги перед конкурентами, які повністю відповідають потребам цільової аудиторії. Додаток реалізований у вигляді WPF застосунку, що обраховує оптимальне розташування сенсорів на

парковці, а це є досить універсальним способом для подальшого розширення додатку новим функціоналом.

Далі у табл. 4.19 проілюстрована трирівнева маркетингова модель товару: уточняється ідея продукту та/або послуги, його фізичні складові, особливості процесу його надання.

Таблиця 4.19 - Опис трьох рівнів моделі товару

Рівні товару	Сутність та складові		
I. Товар за задумом	Веб-сервіс, що надає доступ до сховища триплетів за допомогою HTTP запитів, дозволяє працювати зі SPARQL-ендпойнтом та надає можливості логічного виведення та хмарного розгортання		
II. Товар у реальному виконанні	Властивості/характеристики	М/Нм	Вр/Тх /Тл/Е/Ор
	1. Наявність зручного інтерфейсу користувача	1.Нм	1.Технологічна
	2. Можливість завантаження схеми з XML чи JSON файлу	2.Нм	2.Технологічна
	3. Можливість отримати рішення в форматі XML	3.Нм	3.Технологічна
	4. Можливість змінювати параметри мережевих пристроїв	4.Нм	4.Технологічна
	Якість: згідно до стандарту ISO 4444 буде проведено тестування		
Маркування відсутнє			
Моя компанія: “Parking Solutions”			
III. Товар із підкріпленням	1-місячна пробна безкоштовна PRO версія		
	Постійна підтримка для користувачів		
За рахунок чого потенційний товар буде захищено від копіювання: наш сервіс буде захищений ліцензією.			

Було описано три рівні моделі товару, з чого можна зробити висновок, що основні властивості товару у реальному виконанні є нематеріальними та технологічними. Також було надано сутність та складові товару у задумці та товару з підкріпленням.

Після формування маркетингової моделі товару слід особливо відмітити – чим саме проект буде захищено від копіювання. У даному випадку найбільш вірогідним гарантом буде ліцензія.

Наступним кроком є визначення цінових меж, якими необхідно керуватись при встановленні ціни на потенційний товар (остаточне визначення ціни відбувається під час фінансово-економічного аналізу проекту), яке передбачає аналіз ціни на товари-аналоги або товари субституту, а також аналіз рівня доходів цільової групи споживачів (табл. 4.20). Аналіз проводиться експертним методом.

Таблиця 4.20 Визначення меж встановлення ціни

<i>№ п/п</i>	<i>Рівень цін на товари- замінники, грн.</i>	<i>Рівень цін на товари-аналоги, грн.</i>	<i>Рівень доходів цільової групи споживачів, грн.</i>	<i>Верхня та нижня межі встановлення ціни на товар/послугу, грн.</i>
1.	35000	55000	250000	40000-55000

Наступним кроком є визначення оптимальної системи збуту, в межах якого було прийняте рішення (табл. 4.20):

- проводити збут власними силами і залучати сторонніх посередників.
- користуватися однорівневим каналом збуту;

Таблиця 4.21 Формування системи збуту

<i>№ п/п</i>	<i>Специфіка закупівельної поведінки цільових клієнтів</i>	<i>Функції збуту, які має виконувати постачальник товару</i>	<i>Глибина каналу збуту</i>	<i>Оптимальна система збуту</i>
1.	Придбання підписки та оплата щомісячних внесків для продовження ліцензії	Продаж	Однорівневий	Власні сили та через посередників

Отже, система приносить прибуток завдяки щомісячним внескам для продовження ліцензії та придбанням підписок, продаж застосунку буде виконуватись напряду або через одного посередника.

Останньою складовою маркетингової програми є розроблення концепції маркетингових комунікацій, що спирається на попередньо обрану основу для позиціонування, визначену специфіку поведінки клієнтів (табл. 4.22).

Таблиця 4.22 Концепція маркетингових комунікацій

<i>№ п/ п</i>	<i>Специфіка поведінки цільових клієнтів</i>	<i>Канали комунікацій, якими користуються цільові клієнти</i>	<i>Ключові позиції, обрані для позиціонування</i>	<i>Завдання рекламного повідомлення</i>	<i>Концепція рекламного звернення</i>
1.	Придбання ліцензії на користування застосунком, щомісячне її продовження, користування застосунком на власних серверах.	Соціальні мережі, електронна пошта, мобільні телефони	Наявність зручного інтерфейсу користувача, можливість завантаження схеми з XML чи JSON файлу, можливість отримати рішення в форматі XML і JSON, можливість змінювати параметри мережевих пристроїв	Показати переваги застосунка, у тому числі і перед конкурентами	Короткий ролик з використанням, реклама.

Отже, в табл. 4.22 наведено концепцію маркетингових комунікацій, було визначено, що придбання ліцензії на користування буде здійснюватися в мережі Інтернет, необхідним буде щомісячне її продовження, користування застосунком можливе на власних серверах. Описано ключові пункти, що характеризують наш проект. Наведено завдання рекламного повідомлення та концепцію рекламного звернення.

4.6 Висновки

В даному розділі було проведено аналіз програмного продукту у якості стартап проекту. Можна зазначити що у проекту є можливість комерціалізації, адже ринок надання послуг в мережі інтернет з використанням рекомендаційних систем динамічно розвивається, створюються нові додатки які, в свою чергу, стимулюють попит на різноманітні допоміжні засоби для пришвидшення роботи та оптимізації алгоритмів та матеріального забезпечення.

Було проведено аналіз ризиків та можливостей які можуть виникнути. Основними загрозами, очікувано, виявились конкуренція та зміна потреб користувачів. Найбільш вдалимими можливостями для нас, звичайно ж, є невдачі наших конкурентів. Також гарною можливістю для росту є загальне «підняття» ринку.

Для успішного виконання проекту необхідно реалізувати програму із використанням засобів .Net, WPF, ASP Net. Для успішного виходу на ринок у продукту повинні бути наступні характеристики:

- наявність зручного інтерфейсу користувача
- можливість завантаження схеми з XML чи JSON файлу
- можливість отримати рішення в форматі XML і JSON
- можливість змінювати параметри мережевих пристроїв

На ринку наявна нецінова конкуренція, існує декілька фірм-конкурентів, але всі вони покривають лише якусь певну частину функціональності нашої системи, тому вихід на нього буде потребувати певних зусиль та капіталовкладень. Проте проект є доволі конкурентоспроможним завдяки своїй нижчій собівартості та значно більшій кількості функціоналу. Через те, що він є повністю програмним, його розробка не потребує витрат на різноманітні матеріали та обладнання, необхідні для виготовлення корпусу, схем, тощо.

В рамках даного дослідження були розраховані основні фінансово-економічні показники проекту, а також проведений менеджмент потенційних ризиків. Проаналізувавши отримані результати, можна зробити висновок, що подальша імплементація є доцільною.

ВИСНОВКИ

Під час написання даної магістерської дисертації було ознайомлено із загальною проблематикою Smart Parking, проаналізовано елементну базу сенсорів, сформовано задачу оптимального позиціонування сенсорів, розроблено алгоритм для вирішення задачі оптимального позиціонування сенсорів та на його основі створено програму.

Проаналізувавши існуючі рішення та вивчивши сенсорну базу (розділ №2) було прийняте рішення зробити основним об'єктом оптимізації бездротові Bluetooth комутатори, так як для точного визначення присутності автомобіля чи іншого транспортного засобу на парковці необхідно розмістити датчик присутності безпосередньо на місці для паркування. Такий підхід дозволяє точно визначати зайнятість місця при будь-яких погодних умовах та інших завадах.

Також, окремим розділом, було розділом було коротко розглянуто існуючий прототип системи Smart Parking в Алжирі. Було проаналізовано особливості топології бездротової мережі цієї системи та опис алгоритму, що використовувався для оптимального розташування комутуючих вузлів на цій парковці.

Тепер давайте перейдемо до розгляду програмного продукту, розробленого в ході виконання даної роботи.

В ході роботи над даною магістерською дисертацією було розроблено програмний продукт, що включає в себе реалізацію алгоритму для вирішення задачі оптимального розташування комутуючих вузлів на парковці. Програма є універсальною та вирішує необхідну задачу для будь-якої схеми паркувального майданчика. Схема паркувального майданчика задається за допомогою JSON чи XML файлу з описом майданчика в необхідному форматі. Також програма дозволяє налаштовувати параметри мережевих пристроїв під конкретні цілі, тобто можна

вказувати кількість можливих підключень до комутатора та дальність бездротового зв'язку.

Розроблена програма надає більш широкі можливості для обрахунку оптимального позиціонування датчиків на автостоянці ніж існуюче рішення, що використовувалося при розгортанні прототипу системи Smart Parking в Алжирі, адже програма дозволяє завантажувати довільну схему парковки, налаштовувати параметри мережевих пристроїв та враховує перешкоди для сигналу на парковці. Програма, що використовувалась при розгортанні прототипу в Алжирі основним недоліком мала відсутність можливості змінити схему парковки та вона не враховувала перешкоди для сигналу на самій парковці.

Також, в межах написання роботи, було зроблено ряд тестів для програми з метою перевірки адекватності та оптимальності роботи алгоритму. Тести показали хороше співвідношення швидкодії до оптимальності результату:

- Для парковки на 155 місць з неоднорідною для розповсюдження сигналів проникністю підрахунок результату тривав 37 секунд(табл. 3.2)
- Для парковки на 68 місць з простішою схемою підрахунок результату тривав 10 секунд(табл. 3.1)

Також описано розроблену програму, що складається з двох проектів, один це динамічно приєднувана бібліотека з основними обчисленнями, а інший слугує для відображення результатів обрахунків.

Наведено розділ з аналізом ринку та підготовкою стратегії для створення стартап-проекту.

Отже, було проведено дослідження існуючих рішень для оптимізації топології мережі в системах Smart Parking, розроблено алгоритм для вирішення цієї задачі на парковках з довільним ландшафтом, розроблено програму на основі даного алгоритму та проведено тестування цієї програми.

ПЕРЕЛІК ПОСИЛАНЬ

1. Li, X.; Ranga, U.K. Design and implementation of a digital parking lot management system. *Technol. Interface J.* 2009, 10.
2. Jian, M.-S.; Yang, K. S.; Lee, C.-L. Modular RFID parking management system based on existed gate system integration. *WSEAS Trans. Syst.* 2008, 7, 706–716.
3. Pala, Z.; Inanç, N. Smart parking applications using RFID technology. In Proceedings of the 1 st RFID Eurasia Conference, Istanbul, Turkey, 5–6 September 2007; pp. 121–123.
4. Yass, A.A.; Yasin, N.M.; Zaidan, B.B.; Zeiden, A.A. New design for intelligent parking system using the principles of management information system and image detection system. In Proceedings of the 2009 International Conference on Computer Engineering and Applications, Manila, Philippines, 6–8 June 2011; Volume 2, pp. 360–364.
5. Bong, D.B.L.; Ting, K.C.; Lai, K.C. Integrated approach in the design of car park occupancy information system (COINS). *IAENG Int. J. Comput. Sci.* 2008, 35, 7–14.
6. Tang, V.W.S.; Zheng, Y.; Cao, J. An intelligent car park management system based on wireless sensor networks. In Proceedings of the 2006 1st International Symposium on Pervasive Computing and Applications, Urumqi, China, 3–5 August 2006; pp. 65–70.
7. Bi, Y.; Sun, L.; Zhu, H.; Yan, T.; Luo, Z. A parking management system based on wireless sensor network. *Acta Autom. Sin.* 2006, 32, 968–977.
8. Benson, J.P.; O'Donovan, T.; O'Sullivan, P.; Roedig, U.; Sreenan, C.; Barton, J.; Murphy, A.; O'Flynn, B. Car-park management using wireless sensor networks. In Proceedings of the 2006 31st IEEE Conference on Local Computer Networks, Tampa, FL, USA, 14–16 November 2006 ; pp. 588–595.
9. Geng, Y.; Cassandras, C.G. A new 'smart parking' system infrastructure and implementation. *Procedia-Soc. Behav. Sci.* 2012, 54, 1278–1287.
10. Lu, R.; Lin, X.; Zhu, H.; Shen, X. SPARK: A New VANET-based Smart Parking Scheme for Large Parking Lots. In Proceedings of IEEE INFOCOM 2009, Rio de Janeiro, Brazil, 19–25 April 2009 ; pp. 1413–1421.
11. Samaras, A.; Evangeliou, N.; Arvanitopoulos, A.; Gialelis, J.; Koubias, S.; Tzes, A.

- Novel Smart Parking System based on Wireless Sensor Networks. In Proceedings of the 1st International Virtual Conference on Intelligent Transportation Systems, Slovakia, 26–30 August 2013; pp. 140–145.
12. Chinrungrueng, J.; Dumnin, S.; Pongthornseri, R. IParking: A parking management framework. In Proceedings of the IEEE 11th International Conference on ITS Telecommunications (ITST), St. Petersburg, Russia, 23–25 August 2011; pp. 63–68.
 13. Gu, J.; Zhang, Z.; Yu, F.; Liu, Q. Design and implementation of a street parking system using wireless sensor networks. In Proceedings of the 10th IEEE International Conference on Industrial Informatics (INDIN), Beijing, China, 25–27 July 2012; pp. 1212–1217.
 14. Nawaz, S.; Efstratiou, C.; Mascolo, C. Parksense: A smartphone based sensing system for on-street parking. In Proceedings of the 19th Annual International Conference on Mobile Computing and Networking, Miami, FL, USA, 30 September–4 October 2013; pp. 75–86.
 15. Rose, R. A smart technique for determining base-station locations in an urban environment. *IEEE Trans. Veh. Technol.* 2001, 50, 43–47.
 16. Han, J.K.; Park, B.S.; Choi, Y.S.; Park, H.K. Genetic approach with a new representation for base station placement in mobile communications. In Proceedings of the 2001 IEEE Vehicular Technology Conference, Atlantic City, NJ, USA, 7–11 October 2001; Volume 4, pp. 2703–2707.
 17. Meunier, H.; Talbi, E.; Reininger, P. A multiobjective genetic algorithm for radio network optimization. In Proceedings of the 2000 Congress on Evolutionary Computation, La Jolla, CA, USA, 16–19 July 2000; Volume 1, pp. 317–324.
 18. Amaldi, E.; Capone, A.; Malucelli, F.; Signori, F. UMTS radio planning: Optimizing base station configuration. In Proceedings of the 2002 IEEE Vehicular Technology Conference, Vancouver, BC, Canada, 24–28 September 2002; Volume 2, pp. 768–772.
 19. Church, R.; Velle, C.R. The maximal covering location problem. *Pap. Reg. Sci.* 1974, 32, 101–118.
 20. Mehzer, A.; Stulman, A. The maximal covering location problem with facility placement on the entire plane. *J. Reg. Sci.* 1982, 22, 361–365.

21. Bulusu, N.; Heidemann, J.; Estrin, D. Adaptive beacon placement. In Proceedings of the 21st International Conference on Distributed Computing Systems, Mesa, AZ, USA, April 2001 ; pp. 489–498.
22. Chakrabarty, K.; Iyengar, S.S.; Qi, H.; Cho, E. Grid coverage for surveillance and target location in distributed sensor networks. *IEEE Trans. Comput.* 2002, *51*, 1448–1453.
23. Dhillon, S.S.; Chakrabarty, K.; Iyengar, S.S. Sensor placement for grid coverage under imprecise detections. In Proceedings of the Fifth International Conference on Information Fusion, Annapolis, MD, USA, 8–11 July 2002; Volume 2, pp. 1581–1587.
24. Howard, A.; Mataric, M.J.; Sukhatme, G.S. An incremental self-deployment algorithm for mobile sensor networks. *Auton. Robots* 2002, *13*, 113–126.
25. Howard, A.; Mataric, M.J.; Sukhatme, G.S. Mobile sensor network deployment using potential fields: A distributed, scalable solution to the area coverage problem. *Distrib. Auton. Robot. Syst.* 2002, *5*, 299–308.
26. Zou, Y.; Chakrabarty, K. Sensor deployment and target localization based on virtual forces. In Proceedings of the Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2003), San Francisco, CA, USA, 30 March–3 April 2003 ; Volume 2, pp. 1293–1303.
27. Fonseca, C.M.; Fleming, P.J. Genetic algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization. In Proceedings of the Fifth International Conference on Genetic Algorithms, Urbana-Champaign, IL, USA, 17–22 July 1993; Volume 93, pp. 416–423.
28. Jourdan, D.B.; de Weck, O.L. Layout Optimization for a Wireless Sensor Network Using a Multi-Objective Genetic Algorithm. In Proceedings of the 2004 59th IEEE Vehicular Technology Conference, Milan, Italy, 17–19 May 2004; Volume 5, pp. 2466–2470.
29. Jourdan, D.B.; de Weck, O.L. Multi-Objective Genetic Algorithm for the Automated Planning of a Wireless Sensor Network to Monitor a Critical Facility. In Proceedings of the SPIE, Sensors, and Command, Control, Communications, and Intelligence (C3I) Technologies for Homeland Security and Homeland Defense III, Orlando, FL, USA, 12–16 April 2004; Volume 5403, pp. 565–575.

30. Jourdan, D.B.; Roy, N. Optimal sensor placement for agent localization. *ACM Trans. Sens. Netw. (TOSN)* 2008, 4, 13:1–13:40.
31. Li, W.; Cassandras, C.G. A minimum-power wireless sensor network self-deployment scheme. In Proceedings of the 2005 IEEE Wireless Communications and Networking Conference, New Orleans, LA, USA, 13–17 March 2005; Volume 3, pp. 1897–1902.
32. Biagioni, E.S.; Sasaki, G. Wireless sensor placement for reliable and efficient data collection. In Proceedings of the 36th Annual Hawaii International Conference on System Science, Big Island, HI, USA, 6–9 January 2003.
33. Zhang, H.; Hou, J.C. On the upper bound of α -lifetime for large sensor networks. *ACM Trans. Sens. Netw. (TOSN)* 2005, 1, 272–300.
34. Dhillon, S.S.; Chakrabarty, K. Sensor placement for effective coverage and surveillance in distributed sensor networks. In Proceedings of the 2003 IEEE Wireless Communications and Networking Conference, New Orleans, LA, USA, 16–20 March 2003; Volume 3, pp. 1609–1614.