



**БЪЛГАРСКА АКАДЕМИЯ НА НАУКИТЕ
ИНСТИТУТ ПО ИНФОРМАЦИОННИ И КОМУНИКАЦИОННИ
ТЕХНОЛОГИИ**

Теодора Огнянова Митева

**Алгоритми за групово управление на работи при
въздействие на различни комуникационни връзки**

ДИ С Е Р Т А Ц И Я

за придобиване на образователната и научна степен „доктор“

по специалност 02.01.52 “Роботи и манипулатори”

в професионално направление 5.3. “Комуникационна и компютърна
техника”

Научен ръководител: доц. д-р Димитър Карастоянов

София, 2012 г.

Дисертацията е обсъдена и допусната до защита на разширено заседание на секция “Вградени интелигентни технологии” на ИИКТ-БАН, състояло се на 19.03.2012 г.

Дисертацията съдържа 143 стр., 52 фигури, 9 таблици и 10 стр. литература, включваща 60 заглавия.

Защитата на дисертацията ще се състои на 2012 г. от часа в зала на на открито заседание на научно жури в състав:

Проф. д-р Тодор Нешков

Доц. д-р Пенчо Венков

Доц. д-р Любен Клочков

Доц. д-р Димитър Карастоянов

Доц. Михаил Михов

Материалите за защитата са на разположение на интересуващите се в на ИИКТ, ул. “Акад. Г.Бончев”, бл. 2

2012

Теодора Митева

Всички права запазени

Тази дисертация посвещавам на моето семейство. На майка ми, която неотлъчно бе до мен мен и ме подкрепяше във всяка стъпка, на баща ми без който също нямаше да се справя.

Благодаря

Благодарности

“Make everything as simple as possible, but not simpler”

Алберт Айнщайн

Имах честта да работя под ръководството на доц. Димитър Карастоянов, който ме подкрепяше и помагаше през годините - сърдечно благодаря за помощта.

Също така искам да благодаря на екипа на проф. Мартиноли, който ме прие в лабораторията си и ми помогна в работата ми. Благодаря Ви Николаус, Крис, Джим ! Без вас нямаше да се справя.

Разбира се бих искала да изкажа благодарности и на екипа в лабораторията по Автоматика и Контрол в Гранобъл, без които работата ми нямаше да е завършена.

XXXXXXXXXX

Сърдечно Благодаря на всички !!!

Увод

Микророботите са резултат от нарастващата активност на изследвания на границата между роботиката и микротехнологиите. Днес могат да бъдат реализирани работи с размери до няколко куб. сантиметра. Както конвенционалните работи, микророботите представляват комплексни системи, които съдържат множество различни сензори и изпълнителни механизми и се характеризират с:

- програмируемо поведение (като промишлените работи),
- адаптивност към непредвидимо обкръжение (като съвременните изследователски и сервизни работи),
- дистанционно управление (като телеуправляеми работи).

Фактически основната разлика между макро- и микророботите освен размерите са обектите и областите на приложение.

Микророботите повишават човешките възможности в микросвета.

Управлението на мобилни микророботи може да се извърши както в реално време така и офлайн.

Особеното при системите, работещи в реално време е, че обработката на информацията и управлението се извършват в зависимост от външни събития, настъпващи в предварително неизвестни моменти от

време. Реакцията на всяко събитие включва последователност от процедури, които се изпълняват в строги интервали от време. Системите по принцип могат да работят дълго време автономно без външна намеса (за приемане на данни или подаване на задание). При работа в реално време навременната реакция е по-важна от ефективното натоварване на системата.

За да може да се възползваме напълно от всички възможности на мобилните микро роботи е важно да имаме добре структурирани алгоритми и поставени задачи.

С настоящата работа прилагам алгоритми за групово управление на мобилни микро работи както в реално време така и офлайн. Разработени са алгоритми за обхождане на повърхност, разгледани са случай при които група от роботи изпълняват поставени задачи, синхронизация на роботи в реално време и комуникацията помежду им.

С Ъ Д Ъ Р Ж А Н И Е

Увод.....	6
1 Глава 1 Поставени Задачи.....	10
2 Глава 2 Въведение.....	11
2.1 Разпределено управление на мобилни микро роботи	11
2.1.1 Обзор и анализ на видове среди и приложението на мобилни микро роботи	12
2.1.2 Автономна работа и групово управление.....	22
2.1.3 Групово управление на микророботи - подходи и проблеми.....	29
2.1.4 SWARM роботи	32
2.1.5 Принос към дисертацията: Разработване на алгоритми за проследяване на повърхност.....	33
2.1.6 Wireless управление на група от микро роботи в реално време.....	34
2.1.7 Принос към дисертацията: Разработване на алгоритми за комуникация между мобилни микро роботи	36
3 Глава 3 Обхождане на повърхност.....	37
3.1 Проблеми възникващи при обхождане на повърхност	38
3.2 Граф представяне на обхождане на повърхност.....	44
3.2.1 Определяне на задължителни криви за инспекция - E_R	48
3.2.2 Определяне на криви, предоставящи връзки - E_C	48
3.3 Граф алгоритъм за обхождане на повърхност	49
3.4 Разработване на работна среда за симулация	59
3.5 Резултати.....	74
3.5.1 Видове Ситуации	80

4	Глава 4 Алгоритми за групово управление на мобилни микро роботи във реално време	83
4.1	Безжични комуникационни протоколи.....	84
4.2	Проблеми възникващи при работа в реално време	103
4.3	Опростен модел на управление на два мобилни микро робота	106
4.4	Модел на комуникация на два робота при наличие на шум.....	119
5	Глава 5 Заключение	129
6	Литература	131
7	Таблица на използвани фигури	140

1 Глава 1 Поставени Задачи

В предоставената работа са разгледани проблеми свързани с адаптивно управление на мобилни микро роботи. Поставените цели в този труд е да се намерят рационални алгоритми за обхождане на дадена повърхност и комуникация между отделните роботи.

Поставени цели в разгледаната работа:

- Намиране на рационален алгоритъм за обхождане на пространство.
- Намиране на рационален алгоритъм за пълно и цялостно обхождане на повърхност.
- Намиране на подходящ и рационален начин на комуникация между разгледаните вид мобилни микро роботи.
- Валидиране на алгоритмите за пълно обхождане на повърхност.
- Тестване на всички алгоритми в реална среда.

2 Глава 2 Въведение

2.1 Разпределено управление на мобилни микро роботи

През последните няколко години, принципите за разпределено управление се прилагат успешно в различни области на изследване и проучвания: съвкупност от роботи [1][2][3] и самостоятелни роботи [4], кооперативно транспортиране на обекти [5][6][7][8], съгласуване при разработване на различни стратегии, състезания, обхождане на повърхност [9][10][11][12][13][14][15][16][17]. Всяко едно от тези действия се извърша при независимо взето решение от страна на робота.

Ако се вземе предвид че всяка една поставена задача може да се представи като граф и всяка точка от този граф да бъде с точно определени координати, бихме могли да въведем по прицизни данни на робота. По този начин се спестява време поради това, че на всеки робот се задава различна информация и координати и при група от роботи не се получава припокриване.

2.1.1 Обзор и анализ на видове среди и приложението на мобилни микро роботи

Промислените роботи, предназначени за индустриална експлоатация в конвенционални производства, обикновено са стационарни и работят в сравнително благоприятна среда с предварително известни и описани параметри на обектите. Ето защо основните изисквания към тях са за сила, точност, повтаряемост, [43]. При мобилните роботизирани системи основното предназначение е изследователска дейност, повишаване качеството на живота или работа в неконвенционални и вредни условия. Често това се съпътства с действия в априори неизвестна, неподредена или променяща се среда. Понякога крайният ефект е замърсяване, дефектиране или разрушаване на работа. Всичко това поставя специфични изисквания към конструкцията, управлението, комуникацията, а също и цената на мобилната роботизирана система, [44].

Различаваме видове вредни среди според фактора на вредност, времетраенето на въздействието му, връзката с основната изпълнявана задача, [45]. При стационарни производствени процеси с вредни емисии (изпарения, радиация, светлинно или топлинно излъчване) се прилага изолиране и херметизация на работната зона с работа вътре в нея. Роботите работят автономно на базата на написано програмно

задание, евентуално с елементи на супервайзорен контрол от оператора.

Друг случай са постоянно действащи параметри на средата, правещи невъзможни живота и работата на човек (под вода, в космоса). За целта се ползват миниподводници, луноходи и специализирани манипулатори. Управлението обикновено е командно-дистанционно, а при автономни операции има супервайзорен контрол.

В основните случаи на ползване на мобилни роботизирани системи общите елементи при вредностите са:

- вредности от работната среда – радиоактивно лъчение или химически елементи в атмосферата,
- вредности от терена на придвижване – радиоактивни частици или минирано поле,
- непознат терен на придвижване – разрушения след авария или поначало непознато място,
- наличие на препятствия в зоната – налични машини, отломки, пресечена местност.

Най-общо източниците на вредност независимо от връзката им с работния процес и времето на въздействие могат да бъдат:

- радиоактивно излъчване,

- химическо замърсяване,
- взривоопасни терени,
- взривоопасни обекти,
- силни светлинни и топлинни излъчвания,
- липса на въздух, светлина и топлина.

За случаите на работа в подводни или безвъздушни пространства се вземат мерки за херметизация на корпуса, разсейване на отделяната топлина, специализирани задвижващи органи (перки, дюзи и др.). В общия случай при движение по повърхност тя е непозната и некултивирана. Това налага чупещи се оси при колесни транспортни средства, техническо зрение и командно-дистанционен режим.

При разминирание и антитерористична дейност също основно се работи в командно-дистанционен режим поради многобройните възможни ситуации, опасности и варианти. Освен укрепена конструкция, алтернативен начин тук е модулност и невисока цена на мобилните устройства, поради голямата вероятност от повреда или разрушаване.

За случаите на огледи и разчиствания след аварии в ядрени централи са налице изисквания за широк спектър сензори, измервателни уреди и камери за наблюдение на работната зона, [46].

Поради вероятност от разхвърляни отломки е необходимо наличие в управлението на алгоритми за оценка, избягване или преодоляване на препятствия. При функции на пробовземане е възможна комбинация от носеща платформа с автономно достигане до зададена точка. Върху платформата може да се разположи ръка с командно-дистанционно управление за нуждите на пробовземането и манипулация с предмети в целевата точка.

Различаваме йонизираща (алфа, бета, гама, неутрони и рентгенови лъчи) и нейонизираща (лазерни, инфрачервени, ултравиолетови, радио вълни) радиация. При ядрени аварии следва да се вземат специални мерки (екраниране и др.) за защита на електронните компоненти от лъчения. На етапа на проектиране на системата следва да се подбират внимателно градивните компоненти (метали, композити, хард-рад електроника и др.) с оглед да се осигури работоспособност и да се избегне появяване на вторична радиация след излизане на робота от радиоактивната зона.

За случаи на движение по априори неизвестна траектория в непозната работна среда е за предпочитане безжичната комуникация. При вероятност от смущения на радиокомуникациите може да се

ползва екранирана кабелна връзка за обмен на данни и команди между оператора и мобилната система.

Определения и класификации на микророботи

Микророботите са резултат от нарастващата активност на изследвания на границата между роботиката и микротехнологиите. Днес могат да бъдат реализирани работи с размери до няколко куб. сантиметра. Както конвенционалните работи, микророботите представляват комплексни системи, които съдържат множество различни сензори и изпълнителни механизми и се характеризират с:

- програмируемо поведение (като промишлените работи),
- адаптивност към непредвидимо обкръжение (като съвременните изследователски и сервизни работи),
- дистанционно управление (като телеуправляеми работи).

Фактически основната разлика между макро- и микророботите освен размерите са обектите и областите на приложение. Микророботите повишават човешките възможности в микросвета.

Термините микро и нанороботи обикновено се отнасят до манипулационни системи в рамките на 10 куб см с много висока прецизност. Тук ще приемем, че микророботът е автономно или

полуавтономно устройство с малки размери в рамките на няколко куб. см, което е способно да извършва манипулации в микронната област с резолюция в нанообластта [47]. По-специално мобилните микророботи са голяма група с разнообразни характеристики според предназначение, конструкция, задвижване, скорости, сензорика и др.

Някои класификации на микророботи биват според:

Класификация според размера

Според размера малките работи биват три вида:

- миниатюрни работи: размерите са до няколко куб. см; работното пространство и силите са сравними с манипулации на човешки пръсти; произвеждат се за сглобяване на конвенционални миниатюрни компоненти и микромашини; повечето от наличните днес микророботи са от този клас,
- микророботи: размерите са до няколко стотици куб. μm ; създават се като **средство** за микромашинни технологии (пространствени или равнинни микромашинни операции или литографски - LIGA технологии); съдържат микроманипулатори, сензори, вериги за “сигнал-процесинг”;

примери за ползуване са сглобяване на нанороботи, работа с клетки,

- нанороботи: размерите са до няколко стотици nm, тоест с размерите на човешка клетка; конвенционалните механични принципи за задвижвания и манипулатори не са приложими тук, ползват се електрохимически принципи (като при мимикриращи биологични микроорганизми); твърдотелните технологии за изграждане не са ползуваеми за нанопроизводство, решение могат да бъдат полимерните химични техники.

Функционална класификация

Според функционални и структурни особености говорим за:

- стационарни или мобилни микророботи,
- със собствен или с външен енергоизточник,
- работещи самостоятелно или в група,
- автономни или с дистанционно управление,
- с жична или безжична връзка с оператора,

Класификация според движенията

Според начина на придвижване микророботите биват:

- колесни,
- верижни,
- крачещи,
- катерещи,
- пълзящи,
- летящи,

Класификация според обсега и задачите

Някои автори ползват съотношението “С” между физическите размери на микроробота и работната му зона:

- $C \gg 1$: стационарни микроманипулационни системи,
- $C \ll 1$: микроскопични мобилни роботи.

Задвижването на микророботите трябва да има следните свойства:

- малки размери
- изискващи не голям волтаж
- потенциално производство в единна процедура
- да се сглобява лесно
- да консумира малко енергия

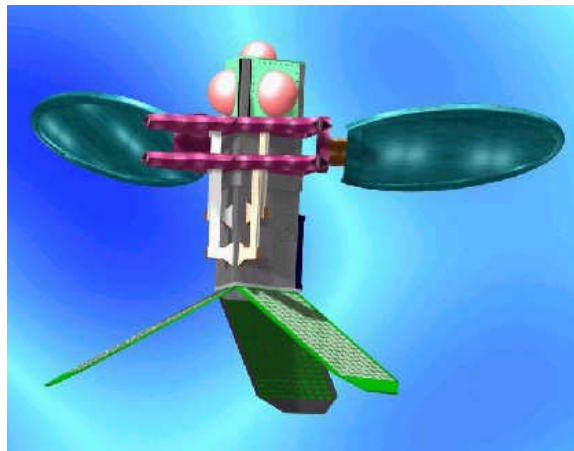
Корпорацията Seiko Epson разработва mFR (микро летящ робот), най-малкият летящ прототип на робот в света – фиг. 1.



фигура 1 Летящ Микроробот mFR

Epson разработва mFR, за да демонстрира собствена микромехатронна технология. Epson разработва и продава семейство от микророботи известни като EMRoS серии, започвайки с Monsieur, който е пуснат на пазара през 1993 г. и е публикуван в книгата на Гинес като най-малкия микроробот на света. През април миналата година Epson разработва Monsieur II-P, микроробот който оперира чрез свръх тънък, свръхзвуков мотор и спестяващ енергия Bluetooth комуникационен модул, който позволява много части едновременно да бъдат контролирани дистанционно и безжично. Използвайки тези роботи, Epson също реализира най-малкия робот за театър и балет. С разработването на mFR Epson демонстрира възможността за разширяване на сферата на микророботите от дву-измерното пространство до три-измерното пространство.

Микромеханичен проект “Летящо насекомо” – Бъркли, САЩ, [48]. Американското правителство е инвестирало \$ 2.5 милиона в проекта на Университета в Бъркли за разработка на роботизирано насекомо с размери за летеж в помещение. Първата голяма стъпка при създаването на това микромеханично летящо насекомо (Micromechanical flying insect – MFI) е създаването на устройството Robofly (фиг. 2.), даващо на изследователите важна информация за механизма на летеж на насекомите



фигура 2 Летящо насекомо Robofly

Институт за управление на процеси и роботика (IPR) – Германия разработва микророботите, предназначени за различни сфери на приложение. Те могат да се разделят на две големи групи: биологични приложения и микромеханика. Биологичните приложения включват

работа с клетки, които имат размери в областта на микроните. Задачите тук са от вида “вземи -постави” за инжектиране на субстанции в клетки (фиг. 3.).



фигура 3 Работа в клетка

Автоматизацията тук включва разпознаване и локализиране на клетките, идентификация на особености (например цветове при ракови клетки, боядисани в червено при сини здрави клетки).

2.1.2 Автономна работа и групово управление

В последните години особено внимание се обръща на така наречените разпределени системи от роботи. При тях поставените задачи се изпълняват не от един робот, а от група от роботи, работещи съвместно. Между членовете на групата се обменя сензорна информация, която определя тяхното колективно поведение.

Групата от роботи определено има някои преимущества по отношение както на възприемането на околната среда, така и при извършване на поставените им задачи. Разпределянето на манипулациите между отделните ѝ членове позволява да се разработват роботи със по-малки размери, с по-малка сложност и естествено по-евтини. От друга страна един робот “вижда” заобикалящата го среда от една гледна точка, докато групата от роботи позволява да се осъществи пространствено възприятие.

Груповото управление на мобилни микророботи предполага нова философия на проектиране. Отпада необходимостта роботът да бъде осигурен със всички необходими подсистеми – сензорна, задвижваща, системи за комуникация, управляваща система. Всеки от членовете на групата сам по себе си може да не разполага с особени възможности, но като цяло “отборът” да е в състояние да осъществява сложни задачи.

Друго преимущество на такива системи е поведението им при повреда на някой от членовете на групата. Докато при единичните роботи повредата довежда до пълна загуба на работоспособността им, при повреда на някой от членовете на екипа се загубват само част от възможностите на системата и тя продължава да функционира, макар и с намалени способности.

При определяне на областите на приложение на такива групи от роботи основен фактор е техният размер. Микророботите, с техните малки размери са изключително ефективни при изпълнение на задачи в ограничени пространства, но притежават и редица недостатъци: по-малък брой сензори, изпълнителни механизми и средства за комуникация, ограничени изчислителни възможности и разполагаема енергия, ниска товароносимост и малка скорост. Развитието на разпределените системи обаче доведе до преодоляване на голяма част от недостатъците им и позволи тяхното използване в по-голям кръг от приложения. Освен за игрови и демонстрационни цели, на такива групи от мобилни микророботи, се възлагат редица задачи в областта на микромехатронните приложения, микромедицински операции, микросглобяване и микротранспорт, както и изследване на околната среда.

Концепция за изграждане на “Общество от роботи”

Общество от роботи (Robot Society - RS) е обширна техническа концепция за описване и управление на група от автономни мобилни роботи, работещи заедно и съобразно с поставена обща цел. Концепцията за RS е въведена от Halme A., Jakubik P., Schönberg T., Vainio M през 1993 [49] и се определя като група от индивидуални

роботи, наречени членове и притежаващи информационна и управляваща структура. Членовете на това общество не е задължително да са еднакви. Тези които имат еднакви възможности или са с еднаква структура, формират клъстери или класове. Информационната структура дефинира начините, по които информацията се “разпространява” между членовете и по които отделните агенти комуникират помежду си. Управляващата структура определя начинът, по който обществото въздейства върху членовете си и управлява изпълнението на общата задача.

Последна практическа цел на RS концепцията е да се създаде такава група от разпределени роботи, които да могат да изпълняват дадена задача, зададена от оператора или от управляващия контролер на “обществото”, както е при конвенционалните индивидуални роботи.

Обикновено управляващата структура на RS е йерархична и се състои от активни и “съвещателни” компоненти. Комуникацията е ключов фактор за правилната работа и може да се раздели на две категории – комуникация между оператора и членовете на обществото (външна) и комуникация между самите роботи (вътрешна).

От инженерна гледна точка, към управляващите архитектури следва да се прилага т.нар. минималистичен подход – т.е. разработване

на максимално проста, но ефективна и лесна за вграждане структура, осигуряваща едновременно както автономното функциониране на всеки член от RS, така и динамичното взаимодействие между тях за осъществяване на кооперативното поведение при изпълнение на поставените задачи. Ако в този аспект интелигентността се разглежда като способност за обмен на информация, то RS притежават интелигентно поведение, особено ако е предвидена възможност за динамична реконфигурация на обществото. Производителността на такава система може да се оценява по редица критерии, например: качествено изпълнение на поставената задача, нейната продължителността, време за безаварийна работа и др.

По подразбиране RS поддържат паралелно изпълнение на поставените задачи. Техническите преимущества на такива хетерогенни системи са:

1. Нивото на редундантност на системата е много високо, което води до висок толеранс по отношение на грешките на системно ниво. Ако част от еднаквите членове на RS се повредят, то във формацията остават достатъчно работоспособни елемента за изпълнението на поставената задача, макар и по-бавно.

2. Размерът на RS, а следователно и неговата ефективност, могат лесно да се променя с намаляване или увеличаване на броя на

роботите, членове на групата. При отворени и децентрализирани структури това води до постигане на голяма гъвкавост, тъй като не се налага преконфигуриране на системно ниво.

3. Колективното използване на няколко робота предполага значително по-проста структура на отделните членове, което от една страна ги прави по-надеждни, а от друга намалява себестойността им и ги прави лесни за масова употреба.

4. При оптимална организация, след като структурата и комуникацията между членовете ѝ са добре настроени за изпълнение на конкретната поставена задача, ефектът от колективното функциониране може да бъде по-голям от този, който би бил при използване на самостоятелно паралелно работещи роботи.

Увеличаването на членовете на RS увеличава неговите възможности, но този ефект е положителен само до определени граници. Ако оптималната численост на системата се надвиши, общата производителност ще спадне поради несъвместимост на ресурсите и работното пространство, т.е. вместо да работят заедно, те ще си “пречат”.

Съзнанието се разбира като съединяване на индивидуалната опитност и включва усещания, мисли, въображение, памет и др. По отношение на животинския свят, съзнанието се определя като способността на организмите да възприемат (и в този смисъл да осъзнават и усещат) избрани признаци на околната среда. По отношение на роботите, най-общо понятието **“съзнание”** обхваща представата или приемането от тях на състоянието на околната среда и изпълнението на зададена работа, както и на вътрешното им състояние и режим. Като Общество от роботи, *колективното “съзнание”* се базира на евентуална предварително зададена информация и на информация, получена от комуникацията между членовете на групата. Като цяло, колективното “съзнание” е субективен опит, но при RS външната супервайзорска система може да определи дали даден “извод” е правилен или грешен, т.е. това обобщеното оценяване се нарича “съзнание” на Обществото от роботи. Ако се разполага и с допълнителна информация от други източници, не само от роботите, лесно може да се оцени производителността на системата като цяло.

Съединяването и комбинирането на информацията от собствените сензори на роботите и от тези на разпределеното общество посредством взаимната комуникация е основен момент при

формирание на действащо съзнание на RS. Съгласуваното колективно съзнание е важен фактор за ефективна и координирана работа.

Като цяло, динамиката и валидността на съзнанието на RS се определя от следните фактори:

- Точност, резолюция, повтораемост и динамичност на данните от сензорите;
- Честота на подаване на данните;
- Статистическа и пространствена надеждност;
- Производителност и обхват на комуникационната мрежа;
- Обработка и синтез на получената информация от различните източници като функция на времето (включително поддържане на “закъсняла” информация);
- Брой на членовете в групата;
- “Осмисляне” на предварителна информация.

2.1.3 Групово управление на микророботи - подходи и проблеми

Реализацията на система от много роботи е значително по-сложна по отношение на навигацията и управлението им, поради необходимостта от прецизна координация между членовете на групата. Основно съществуват два типа архитектури за групи от роботи – централизирана и децентрализирана.

При централизираната архитектура всички задачи по планиране, контрол, наблюдение и изпълнение се осъществяват от едно общо управляващо устройство. Разбира се, изключително трудно е изграждането на напълно централизирана система за управление на множество роботи, поради големия брой изчислителни операции, пропорционално нарастващи с броя на микророботите.

Този недостатък се избягва при децентрализираните (разпределени) архитектури, където отсъства централно управляващо устройства, а решенията се вземат от самите членове на групата. Тези архитектури се реализират на базата на отделни модули, всеки от които има специфична функция.

Един от проблемите при координацията и управлението на група от роботи е локализацията на всеки от членовете на отбора, т.е. определянето на неговото пространствено положение и ориентация. Това налага да се разработват “рамки”, вътре в които се разполагат сензори за локализация, които подават надеждна и последователна информация за определяне на местоположението на микророботите. При децентрализираните архитектури, обработката и оценката на данните от тези сензори също е разпределена между отделните членове

на екипа, с цел намаляване на изчислителната сложност и съответно на времето за обработка на информацията

Друг важен проблем при изграждането на разпределена система от микророботи е тяхното координирано поведение, т.е. управлението им като група, така че да се използват ефективно техните сензорни и задвижващи системи. Напоследък особено популярен стана методът за управление, базиран на поведението на групата. Arkin и Balch [50] описват инспириран от биологията метод за контрол и интегриране на роботите в групата – всеки робот локално се управлява от арбитрирана, реактивна система, която се стреми да балансира много поведения за достигане на локално определени цели, аналогично на някои системи в живите организми.

Parker [51, 52] предлага подобна архитектура, която осигурява “адаптивен избор на поведение”, което може да се “усилва” или “подтиска” от други действия.

Scott M. Thayer и Surya P. N. Singh представят концепция за ефикасно управление на голяма система от микророботи (до 1500 броя), т.нар. “килороботи”, базирана на механизми подобни на тези, с които функционира човешката имунна система - *Immunology- derived Distributed Autonomous Robotics Architecture (IDARA)*, [25].

Предимствата на системи с такъв подход на управление е в това, че се намаляват трудностите при избор на управляващи въздействия чрез намаляването им до обобщаване на векторни величини, отчитани директно от сензорите.

2.1.4 SWARM роботи

Тук се сблъскаме с парадигмата дали е възможно роботи с такива размери да бъдат достатъчно стабилни и в същото време да изпълняват точно зададените задачи. SWARM роботиката взима своето вдъхновение от природата. Следва поведението на пчелите, мравките, червейте [18][19][20].

SWARM роботиката има четири много важни предимства:

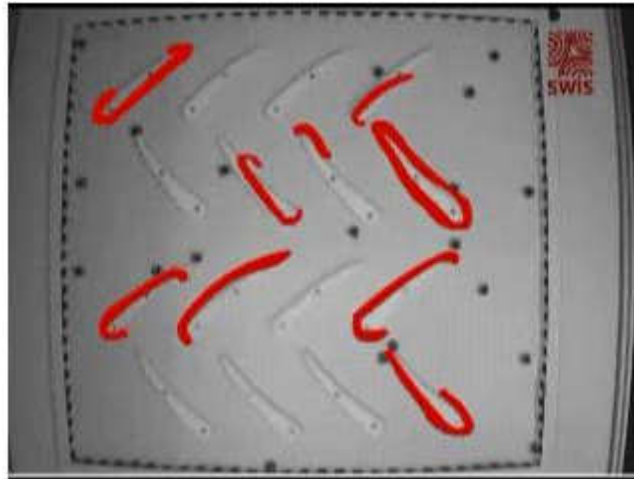
1. Архитектурата при управление може да се поддържа по същия начин както с няколко робота така и при наличието на хиляди роботи.
2. Роботите могат да бъдат прибавени или да се изтеглят от системата по начин по който не се получава противоречие със средата. Те имат способността да се прекоординират и разпределят самостоятелно.

3. Робустни – Въпреки малките си размери тези роботи както бе споменато взимат поведението си от природата. Добре позната структура имат мравките [20].
4. Опростяване – намаляване на общите разходи и минимизиране на единицата

При моделиране на такива системи се взима също предвид и размерите на роботите. SWARM роботите ни позволяват мобилност и с тяхна помощ може да се проникне на трудно достъпни места

2.1.5 Принос към дисертацията: Разработване на алгоритми за проследяване на повърхност

За управлението на мобилни микро роботи една от важните характеристики е да се определят правилните алгоритми. С настоящата работа разглеждам алгоритми за групово управление и разработвам такива с цел цялостно и пълно обхождане на повърхност [21][22][23][24][25][26][27].



Фигура 4

С помощта на теория на графите разглеждам проблема като проблем на китайският пощальон. Правя симулация и алгоритмите са тествани в реално време с роботи тип – e-Ruck. Направила съм анализи и съм приложила резултати

2.1.6 Wireless управление на група от микро роботи в реално време

До скоро безжичните комуникации се свързваха само с приложение в мобилните телефони. Но има и други области, където удобството на безжичните комуникации ги е изтласкало на върха на комуникационната вълна. Безжични мрежи, връзка със специализирани

сензори, предаване на данни, свързани с експлоатационни параметри и още много са областите в които напоследък безжичните комуникации намират приложение. Използването на радиоканалите за връзка между устройствата, макар че не е ново като изобретение, едва напоследък еволюира значително, особено благодарение на широкото разпространение на Internet, локалните и WAN мрежи, свързващи много хора, позволяващи да си разменят глас, видео и данни помежду си, и то с осезаемо високи скорости. Тласък за развитието именно на безжичните устройства е дала необходимостта от по-голяма свобода и удобство при изграждането на мрежи, необходимостта от лесното включване на все по-бързо увеличаващия се брой на мобилни абонати, не желаещи да търсят специални точки за включване към мрежата, а глобално погледнато - потребността на съвременния човек от модерни бързодействащи и високоскоростни комуникации. Според изчисленията на аналитиците, през следващата година в света ще има над 1 милиард мобилни устройства, оборудвани с безжична връзка. Поради тази причина и много от фирмите производители съсредоточават своите усилия в тази област и се разработват (напоследък едночипови) устройства, ползващи един или друг протокол за предаване на данни.

По-подробно са разгледани протоколите Bluetooth и ZigBee.

2.1.7 Принос към дисертацията: Разработване на алгоритми за комуникация между мобилни микро роботи

При среда в която два робота трябва да комуникират по между си от изключително значение е да се зададат правилните алгоритми, както и да се направят правилните конфигурации при комуникацията на роботите, така и на самите роботи.

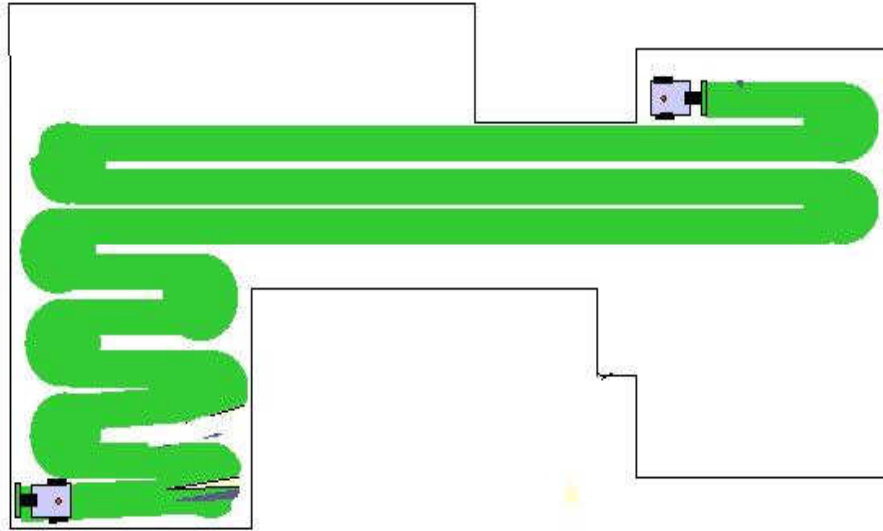
В тази работа разработвам алгоритми за комуникация между два робота. От значение е също така и валидацията на алгоритмите. Системата е тествана в Лабораторията по Автоматика и Контрол в Университета в Гренобъл, Франция

3 Глава 3 Обхождане на повърхност

В тази глава ще бъде разгледан проблема, свързан с обхождането на среда, където група от k робота трябва да инспектират всяка точка от средата в двуизмерна координатна система. Използва се граф презентация на опростен модел, средата. Представени са два метода за конструкцията на графа [28]. Първият метод, обозначен с термина *видим район*, е разгледан в част 3.2, но той води до изключително усложнен граф. Такъв граф не е необходим за поставените цели. Вторият метод *следвайки границите (СГ)* е разгледан в част 3.3 и е резултат на няколко подобрения на първият метод. СГ метода води до много по-задоволителна инспекция на средата, има по добра граф презентация и също така може да поддържа среди съдържащи издути обекти. Графът G , който трябва да се построи, е недиректен напълно свързан граф, съдържащ ъгли E_R и всеки един от тези ъгли трябва да бъде обходен. Такъв тип граф може да се опише с „ k -Rural Postman Problem”, (kRPP) [29]. Роботите използват алгоритми за цялостно обхождане на средата.

3.1 Проблеми възникващи при обхождане на повърхност

Цялостен алгоритъм за обхождане на повърхност е този, който създава път, по който роботът да обхожда дадена повърхност за най-малко време и без да минава по един и същ път два пъти. Въпросът, пред който се изправяме е, как точно да се определи правилния път на робота за изпълнение на задачата. Един от начините е роботът да се движи по права линия, докато достигне препятствие и после да завие. Проблемът при този начин на обхождане обаче е, че може да има места които робота да не достигне и поради тази причина да не обходи



Фигура 5 Примерна среда за обхождане на повърхност, възникващи проблеми

Началната база от която изхождаме е:

- роботите се движат независимо един от друг и хаотично
- разделяне на средата на малки части по начин, такъв че цялата среда да бъде обходена без да се обхожда повторно дадена част

Поставената задача е в оградена дву-измерна затворена среда. В средата има N на брой обекти, $O = \{O_1, \dots, O_N\}$. Предполага се, че повърхността на всеки обект е гладка, затворена, изпъкнала крива и координатите на всеки обект, както и геометрията на затворената среда са известни. Инспекцията ще се направи от k идентични робота, които

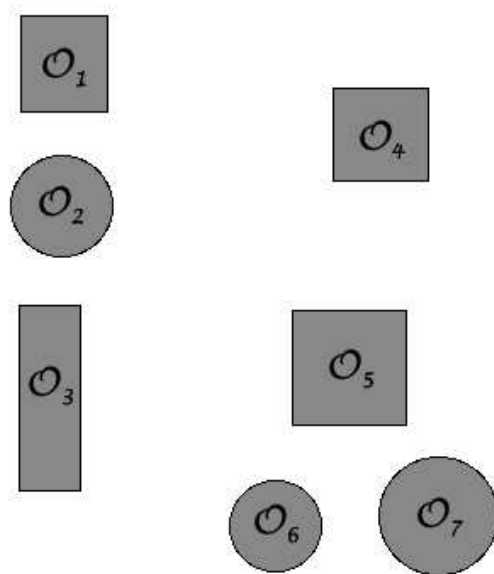
предварително са снабдени със схема и сензори за определение на местонахождението. Сензорите могат да измерят разстоянието лимитирано от $r_{\max}$ до обектите и ограничаващите прегради на средата, в която се намират роботите. Точка p се счита за обходена, когато пътя по който се движи робота обхожда имажинерната права до следваща точка между двете точки p и разстоянието е по малко и равно на $r_{\max}$. Всички роботи които участват в тази инспекция започват от една и съща точка, наречена „*depot*” и се връщат обратно в същата точка когато завършат дадената им задача. Поне веднъж по време на груповата инспекция най-малко един робот трябва да обходи поне една точка от заградената площ. Разпределянето на задачите между роботите се извършва от главен контролен орган. След като получи своята задача всеки робот изчислява маршрута си който е зададен със серия от координати. Всеки робот е конструиран с ограничени комуникационни способности, даващи му възможност да получи своите задачи.

За разработване на маршрут от пътеки, за обхождане, се взима под внимание видимостта на сензора. Видимият район на обект O_j съдържа група от индивидуални позиции в свободното пространство на робота, за които разстоянието до най-близката точка на обекта от средата е по-

малко или равно на $r_{\max}$. Външният периметър на видимата част на всеки обект от средата е крива, която достига разстоянието $r_{\max}$.

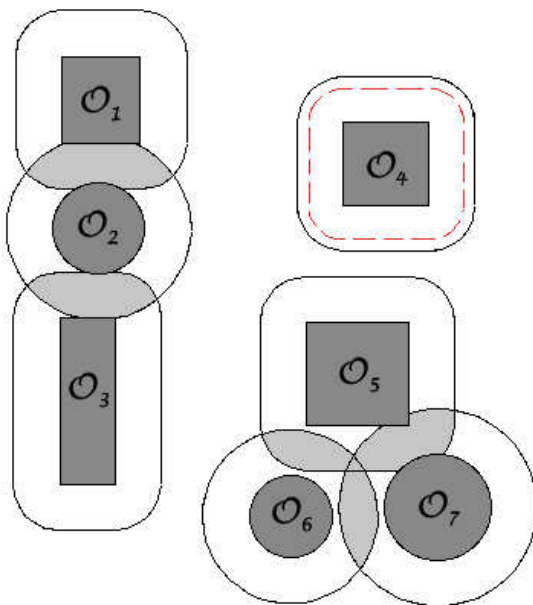
„Сегментна граница” е свързан интервал от ∂O . Сегментните ограничения, които роботите трябва да обхождат се наричат „вътрешни сегментни граници”, обединението на които се задава с ∂O_{int} . Всички останали интервали от ∂O се наричат „външни сегментни граници”, обединението на които се задава с $\partial O_{\text{ext}} = \partial O / \partial O_{\text{int}}$.

На фиг 6 е показана модел на средата.



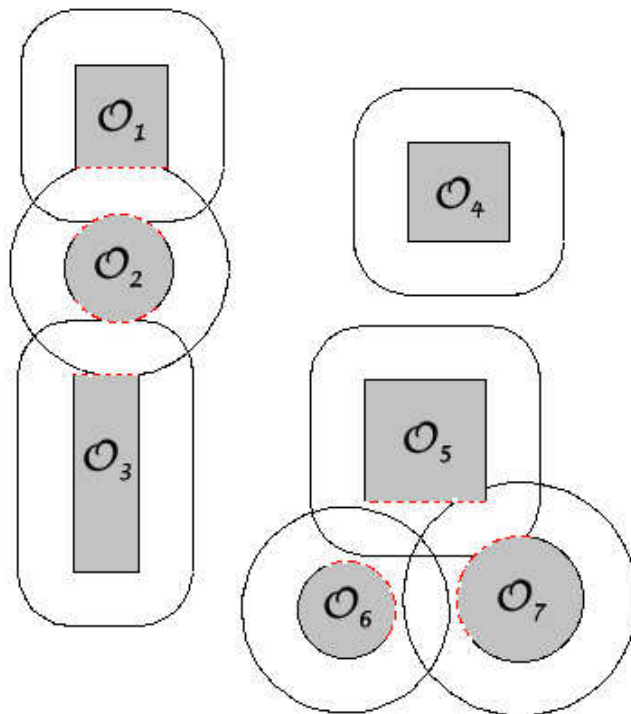
Фигура 6 Работна среда

На фиг. 7. са показани видимите райони на обхождане. С прекъснатата линия е показана видимата част на обект O_4 , възможна за обхождане.



Фигура 7 Области на видими зони за обхождане

На фиг 8. са показани „външните сегментни граници”, за които стана дума по-горе.



Фигура 8 Външни сегментни граници

За целите на задачите които сме си поставили е построен граф, който представя средата за движение на роботите. Графът е разделен на няколко части в зависимост от броя на роботите, които участват в обхождането.

3.2 Граф представяне на обхождане на повърхност

Задължителният маршрут, по които всеки един робот трябва да мине се състои от – задължителните ограничения на всяка точка – E_R , и свързващите ги криви – E_C . Всяка задължителна крива, която робота трябва да инспектира, представлява съвкупност от видими пътища, по която робота трябва да мине, за да завърши дадената задача – покритие на повърхност. Това представяне позволява гъвкавост на навигационните изпълнения, при които роботите се избягват взаимно, като всеки един от тях тръгва по свой път на обхождане. Точките за обхождане са върхове, които физически съответстват на крайни точки. За яснота е представен граф, който лесно определя предпочитаната видима част за инспекция. Използвана е функция, която дава на всяка крива за обхождане тегло, равно на разстоянието за обхождане.

Ориентиран граф $G(V,E)$ се състои от крайно множество от върхове V и множество от (ориентирани) ребра или дъги E . Една дъга е наредена двойка върхове (u,v) и обикновено се означава със стрелка сочеща от u към v . Неориентираните графи могат да се разглеждат като специален вид ориентирани графи, в които, ако има дъга (u,v) , то има и дъга (v,u) . Така посоката на ребрата е без значение и неориентираният граф

$G(V,E)$ се състои от крайно множество от върхове V и множество от ребра E , където всяко ребро е ненаредена двойка върхове $\{u,v\}$.

СИЛНО СВЪРЗАНИ КОМПОНЕНТИ - Два върха u и v в ориентиран граф $G(V,E)$ наричаме свързани, ако има път както от u до v , така и от v до u

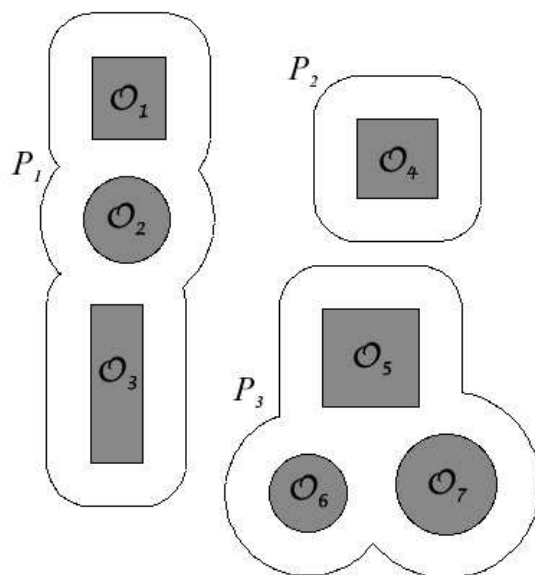
Крайният резултат на описаната конструкция е недиректен свързан граф:

$$G = (V, E, c: E \rightarrow \mathbb{R}^+),$$

където V - група от върхове на дадения граф, E - група от криви за обхождане, а функцията c задава теглото на всяка крива $E_R \dots E$.

Останалите криви $E_C = E/E_R$ са съвкупност от пътища за обхождане, които представляват маршрути за връщане до началната позиция, както и до отдалечени области за инспекция.

За улеснение на дадената задача се предполага, че ограничението за предпочитаната дистанция $r = r_{\max}$. Това ограничение е въведено, за да може да се направи конструкцията на графа по-лесна от тази лимитирана от първоначално определените пътища за обхождане.

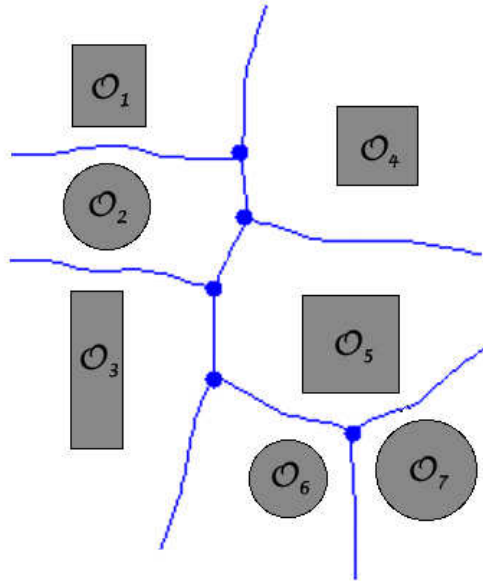


Фигура 9 Райони за инспекция

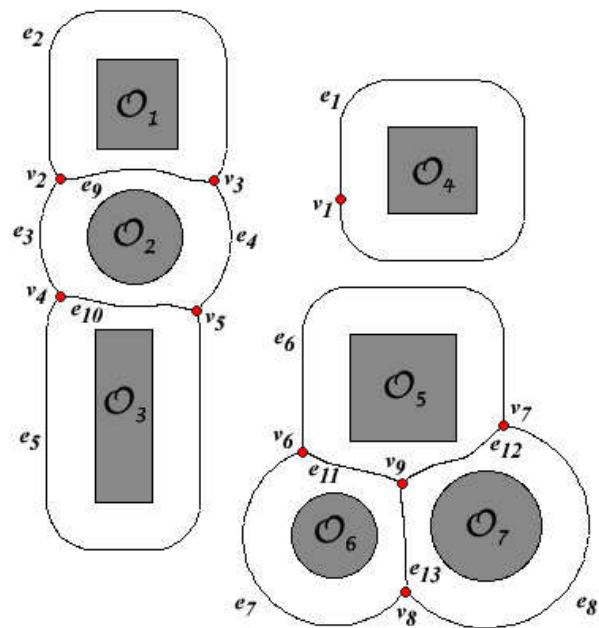
Районите за инспекция на дадена примерна среда са показани на фиг. 9.

Показани са също локални компоненти на GVG (фиг 10)

Конструкцията на задължителните криви за обхождане е представена на
фиг 11



Фигура 10 Локални компоненти



Фигура 11 Конструкция на задължителни криви и точки за обхождане

3.2.1 Определяне на задължителни криви за инспекция - E_R

Нека S_r определя група от криви, които заместват разстояние r за всеки обект в заградената среда. За $r = r_{\max}$, тези криви представляват периметъра на всеки обект във видимата му част, както е показано на фиг. 12. Нека P показва обединението на свободните области P_i , заградени от тези криви. Областта P съдържа една или няколко необединени области P_i , $i \in \{1, \dots, m\}$. Това е илюстрирано на фиг. 9 (3.2a). Областта на P , ∂P , е съставена от m изграждащи затворени криви ∂P_i , $i \in \{1, \dots, m\}$. Всяка една точка в ∂P е създадена така че да се намира в загражденията на областта P .

3.2.2 Определяне на криви, предоставящи връзки - E_c

Групата от задължителни криви за обхождане, асоциирани с инспекцията на затворената среда P_i , формират видим локален граф от G . Компонентите, които е възможно да бъдат отделени от този локален граф, трябва също да бъдат свързани, за да може роботът да обходи цялостно средата.

При така поставената ни задача трябва да се вземе предвид че имаме напълно свързан недиректен граф. Началната точка v_{depot} е свързана с

всяка точка от дадената среда с препятствия. Всяка връзка представлява директен път по който робота може да се движи. На всеки един път се задава разстояние от точка до точка.

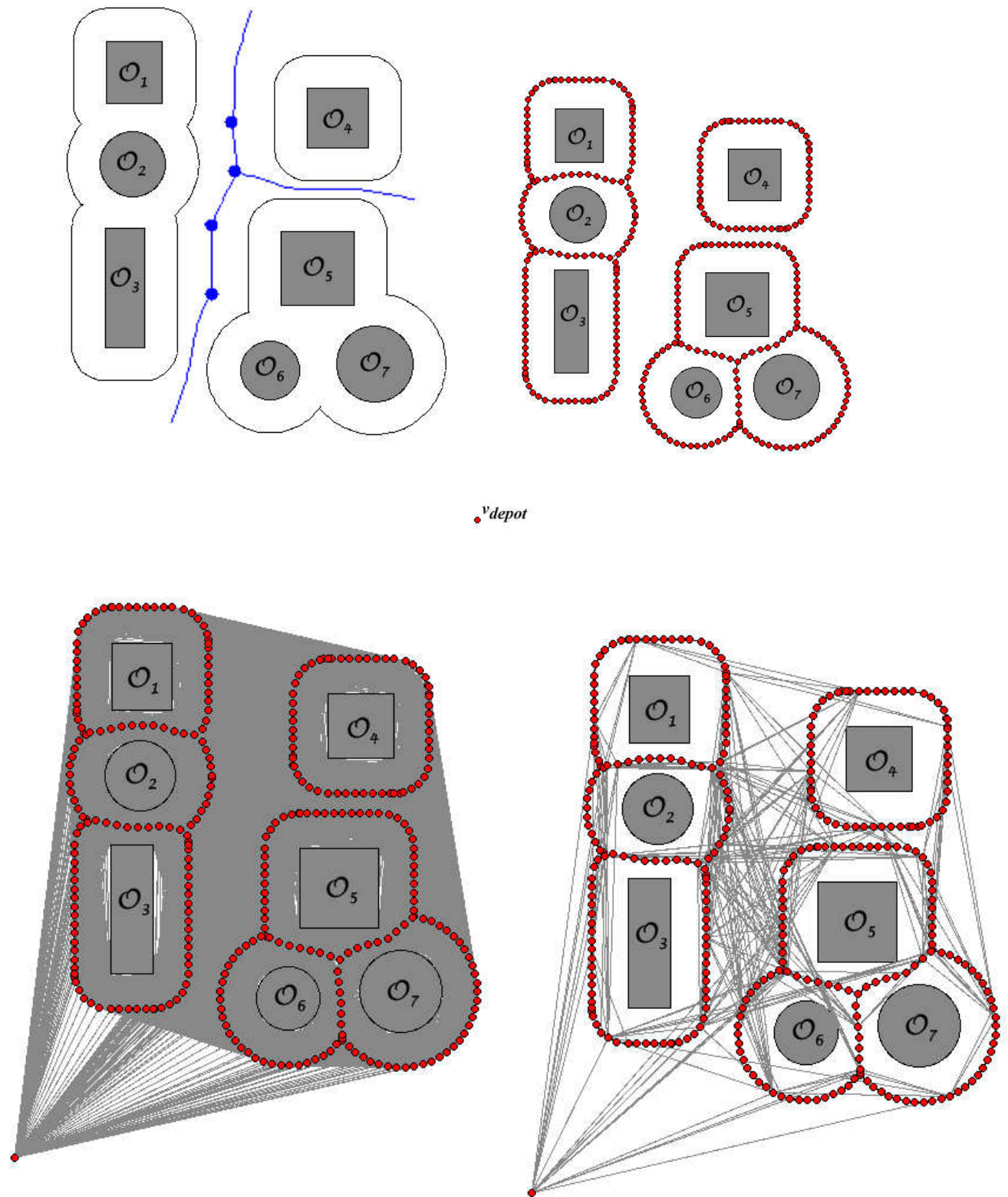
3.2.2.1 Намаляване размера на граф

Тъй като броя на върховете също и броя на пътищата за обхождане става голям и поради тази причина се изисква повече време за пресмятане и предаване на информация, прилагаме метод за намаляване броя на точките за пресмятане.

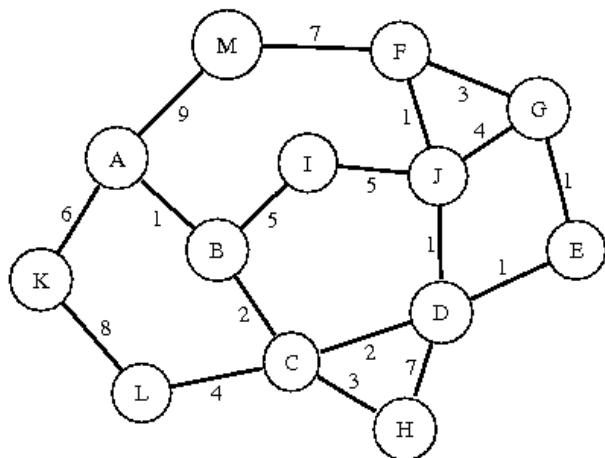
3.3 Граф алгоритъм за обхождане на повърхност

Алгорът е направен с цел да се намери най-краткия път от дадена стартова позиция до всички останали точки от дадената среда за обхождане, където дължината на всяки път, който трябва да се обходи е различна. Имаме директен граф със зададени стойности на всяка линия, **които представят** всеки път, който робота трябва да измине.

На фиг. 13 е показан графа G, който е съставен за средата, която трябва да бъде обходена.



Фигура 12 Презентация на напълно свързан не директен граф



Фигура 13 Прилагане на GraphViz приложение за конструиране на граф във работна среда

Стойността на пътя между два върха е сума от стойностите на пътищата между тези два върха.

За решаване на задачата разглеждам няколко известни алгоритъма:

Алгоритъмът на Дейкстра служи за пресмятане на най-къс път от даден връх до всички останали в ориентиран граф с неотрицателни тегла на ребрата. Може да се модифицира и да се използва за намиране на някои други видове оптимални пътища. Резултатът от алгоритъма е дърво на оптималните пътища от дадения връх до всички останали. Алгоритъмът намира приложение най-вече в информатиката и логистиката, където често се търси най-късото разстояние между две точки (в програма за GPS устройства, където трябва да се планира най-краткото трасе между стартовата и крайната позиция; оптимизация на

транспорта; задача за търговския пътник). Първоначално всеки връх има безкрайна дължина, която се намалява при изпълнение на алгоритъма. Алгоритъмът работи, като поддържа дължината на намерения до момента най-кратък път до всеки връх и постепенно разширява множество от върховете, за които тази дължина със сигурност е оптимална. Алгоритъмът приключва, когато всички върхове на графа са в множеството S .

При Алгоритъма на Прим множеството от ребра X е дърво, а множеството от върхове S съдържа върховете на X . Отначало X съдържа един произволен начален връх от графа. За намирането на най-лекото ребро между S и $V-S$ алгоритъмът поддържа приоритетна опашка, в която се намират всички върхове от $V-S$, които са съседни на някой връх от S . Приоритета на даден връх, според който се построява пирамидата е теглото на най-лекото ребро от този връх до някой връх от S . Това много напомня на Алгоритъма на Дейкстра (където за ключове се използват дължини на пътища вместо тегла на ребра). Както и при Алгоритъма на Дейкстра всеки връх има предшественик $prev[v]$, който е другия край на най-лекото ребро от v до връх от S . Псевдокодът на алгоритъма на Прим е почти идентичен с псевдокода на Алгоритъма на Дейкстра.

При Алгоритъма на Крускал се прилага друга стратегия. Вместо нарастването на едно единствено дърво този алгоритъм се опитва на всяка стъпка да постави най-лекото възможно ребро в дървото. Отначало ребрата се подреждат по нарастващи дължини. Множеството X е празно и върховете на графа се разглеждат като тривиални дървета (без ребра). За всяко ребро от сортирания списък се проверява последователно дали неговите краища са в различни дървета. Ако е така, реброто се добавя към множеството X и двете дървета се обединяват в общо дърво. Множеството S неявно е множеството от върхове на едното от двете обединяващи се дървета.

Методът на Прим може да се характеризира като:

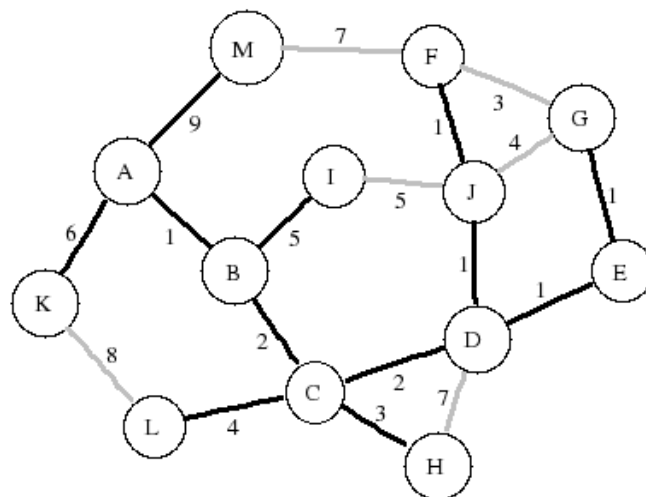
- Можете да се концентрирате само върху едно дърво в даден момент,
- Можете да се концентрирате върху само няколко ръбове в даден момент

Методът на Крускал може да се характеризира като:

- Може да се видят всички краищата наведнъж
- Може да задържат всички върховете, наведнъж
- Може да се разглеждат гори, а не само едно дърво

След направен анализ стигнах до заключението че за поставените задачи, най-рационалният алгоритъм който трябва да използвам е Алгоритъма на Дейкстра, поради това че: 1) дава право да се намери най краткото разстояние от една начална точка до всички останали, а не от всяка една точка до всяка една точка както е при Крускал алгоритъма. 2) не се стреми на всяка следваща стъпка да доваби най-лекото ребро. 3) дава по-лесна реализация на обратния алгоритъм при разделяне на графа на подграфи. 4) по лесни математични операции.

След като веднъж е приложен този алгоритъм се получава най-краткият възможен път от зададена стартова позиция до най-отдалечената точка в средата. На фиг 14 е показан начина, по който алгоритъмът е намерил най-краткия път (фиг 15).



Фигура 14 Приложен Дейкстра алгоритъм за намиране на най-краткото разстояние от първа до последна точка на графа

AA Distances : 0
 AB Distances : 1
 AC Distances : 3
 AD Distances : 5
 AE Distances : 6
 AF Distances : 7
 AG Distances : 7
 AH Distances : 6
 AI Distances : 6
 AJ Distances : 6
 AK Distances : 6

Фигура 15 Прилагане на Дийстра алгоритъм за намиране на най-краткото разстояние от зададена начална точка

След като е приложен алгоритъмът за намиране на най-краткия път се прилага алгоритъмът на Керстин [30], при даден недиректен граф G :

$$G = (V, E, c : E \rightarrow \mathbb{R}^+),$$

и група от върхове u и v , $\{u, v\}$. С V е означено да представя най-краткия път между два върха $SP(u, v)$. Дължината на този път през G е означена с $C_G(SP(u, v))$. Разстоянието между G и всеки един връх v както и линия $e = \{x, y\}$ се намира по следния начин:

$$d(v, e) = \max\{C_G(SP(v, x)), C_G(SP(v, y))\}.$$

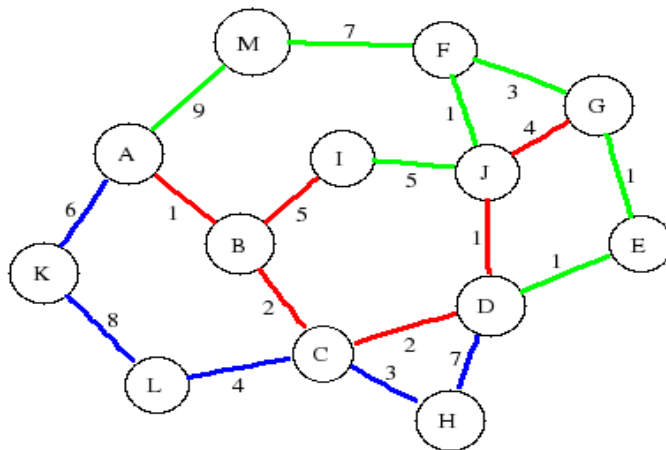
Разстоянието между две линии $e = \{x, y\}$ и $g = \{u, v\}$ на графа G е намерено чрез:

$$d(e, g) = \max\{CG(SP(u, x)), CG(SP(u, y)), \\ CG(SP(v, x)), CG(SP(v, y))\}$$

Алгоритъмът започва с разделянето на графа на исканите линии E_R на k групи за всеки един от роботите – $F_1 \dots F_k = E_R$ като се използва метода за най-отдалечен клъстер, подобен на клъстер алгоритъма за Min-Max К-проблем известен като „Проблема на китайския пощальон”, [31][32], представен от Ар и Райнелт. Методът е базиран на алгоритъм, който има за цел да минимизира максимума на разстоянието до клъстера.

След прилагане на този алгоритъм имаме граф, разделен за три робота (фиг 16).

Всеки един път, оцветен в различен цвят, показва по кой път трябва да тръгне всеки един от роботите. Също така са показани и съответните разстояния и резултати след прилагане на алгоритъма. Алгоритъмът е приложен три пъти за всеки един от роботите.

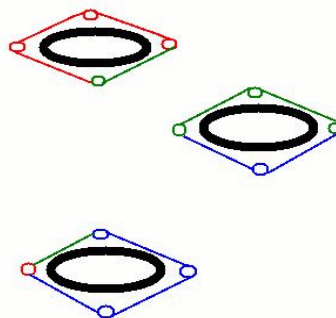


AA Distances 1: 0	MA Distances 2: 9 Sum: 9	LA Distances 3: 9 Sum: 18
AB Distances 1: 1	MB Distances 2: 10 Sum: 11	LB Distances 3: 11 Sum: 22
AC Distances 1: 3	MC Distances 2: 11 Sum: 14	LC Distances 3: 14 Sum: 28
AD Distances 1: 5	MD Distances 2: 9 Sum: 14	LD Distances 3: 14 Sum: 28
AE Distances 1: 6	ME Distances 2: 10 Sum: 16	LE Distances 3: 16 Sum: 32
AF Distances 1: 7	MF Distances 2: 7 Sum: 14	LF Distances 3: 14 Sum: 28
AG Distances 1: 7	MG Distances 2: 10 Sum: 17	LG Distances 3: 17 Sum: 34
AH Distances 1: 6	MH Distances 2: 14 Sum: 20	LH Distances 3: 20 Sum: 40
AI Distances 1: 6	MI Distances 2: 13 Sum: 19	LI Distances 3: 19 Sum: 38
AJ Distances 1: 6	MJ Distances 2: 8 Sum: 14	LJ Distances 3: 14 Sum: 28
AK Distances 1: 6	MK Distances 2: 15 Sum: 21	LK Distances 3: 21 Sum: 42
AL Distances 1: 7	ML Distances 2: 15 Sum: 22	LL Distances 3: 22 Sum: 44
AM Distances 1: 9	MM Distances 2: 0 Sum: 9	LM Distances 3: 9 Sum: 18

Фигура 16 Три пъти приложен алгоритъм за намиране на най краткото разстояние

Следвайки тази теория всеки един робот има вече свой локален граф който да следва, за да завърши дадената му задача. За да може обаче да започне това обхождане ни трябва и координатите на всяка една точка и робот. Тези координати са взети от GraphViz [33] файла.

След като се приложи алгоритъм за намаляване броя на точките за обхождане се получава че всеки обект за да се обходи напълно и изцяло трябва да има четири точки през които робота да премине, фиг 17. Необходимо е да се изтъкне че не е задължително четирите страни на един обект да се обхождат от един робот. Както се вижда от показаната по долу фигура всяка една страна от обекта може да се обходи от различен робот.



Фигура 17 Разпределение на път

3.4 Разработване на работна среда за симулация

Среда или както ще бъде наричано от тук нататък – „свят” е виртуална среда в която се създават обекти. „Свят” се запазва в директория наречена „world” на даденият проект [34]. Запазва се във файл с разширение „.wbt”, който съдържа описание на всеки един обект от средата която е създадена. Този файл не съдържа цялата информация, за да може да се стартира симулация. Контролерът на всеки робот се описва описва във файл и се прави връзка между „.wbt” и файла с контролера.

Роботите използвани в тази работа са – e-puck роботи. В тази глава ще бъде показано как от създаването на елементарен мобилен робот се стига до микро-робот e-puck. Описаният робот се управлява чрез Брайтенберг алгоритъм тъй като целта е да се обхождат препятствия. Първото нещо което трябва да се направи е да се създаде директория в която ще бъде запазен проекта. След това се създават две поддиректории – „worlds” и „controllers”. „Worlds” директорията съдържа всички симулации свързани с робота и

проекта, „controllers” директорията съдържа информация как се управлява роботът. На фиг. 18 е показан робота от който в последствие ще бъде създаден „e-rack” робот с който са правени симулациите в тази работа.



Фигура 18 Първи Webots робот

Средата която се използва за симулации, съдържа под/ основа, препятствия, които да се обхождат и стени, които ограждат средата за обхождане. За всеки нов „свят” се описва осветлението при което се работи. Това става по следния начин:

1. Избира се „PointLight”, от неговите поддиректории се избира – „ambientIntensity”, въвежда се стойност – 0.6. След това се избира „location” и се въвеждат следните стойности – [0.75 0.5 0.5].
2. Създава се още едно „PointLight” и се въвеждат стойности – [-0.5 0.5 -.35] в полето за „location”

3. Създават се още две „PointLight”, като за първото в полето на „location” се въвеждат стойности – [0.45 0.5 -.05], а във второто „PointLight” – [-0.5 0.5 -0.35]

С описаните по-горе три стъпки средата с която ще работим е осветена по желан от нас начин. Понеже работим със затворена среда трябва да се създадат преградни стени които да ограничават желата за обхождане среда. Създаването на стените става по следния начин:

1. Избира се последното „Transform” меню от панела от ляво на средата. Избира се бутона - „insert after”.
2. Избира се „Solid”
3. Отваря се така ново създаденото „Solid” и се пише „wall” в полето
4. Избира се полето „children” и отново се натиска бутона „insert after”, като този път се поставя „Shape”
5. Отваря **се** така създаденото „Shape” поле и се създава възел „Appearance” с бутона „New node”. По същият начин се създава възел „Material” в полето „material” на възела „Appearance”. За избиране на цвят на средата се избира „diffuseColor” от „Material” полето.

6. По същият начин се въвежда цвят на пода.
7. Създава се „Extrusion” възел в полето „geometry” на „Shape”.
8. „convex” се слага като „FALSE”. От тук се задават и координатите на стените на средата. Координатите на стените са както следва:

(x, z) координати:

0 (0.5 , 0.5)

1 (0.5 ,-0.5)

2 (-0.5 ,-0.5)

3 (-0.5 , 0.5)

4 (-0.49, 0.5)

5 (-0.49 , -0.49)

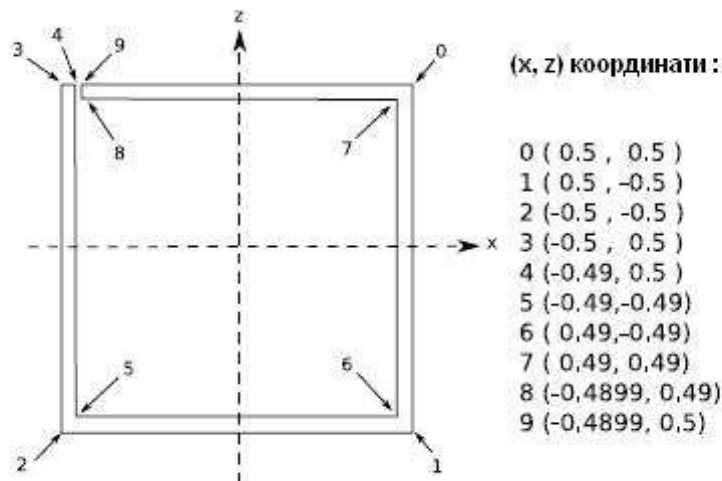
6 (0.49, -0.49)

7 (0.49, 0.49)

8 (-0.4899, 0.49)

9 (-0.4899, 0.5)

На фиг. 19 са показана средата с ограничаващите я стени и съответните координати:



Фигура 19 Страни и координати на работната среда

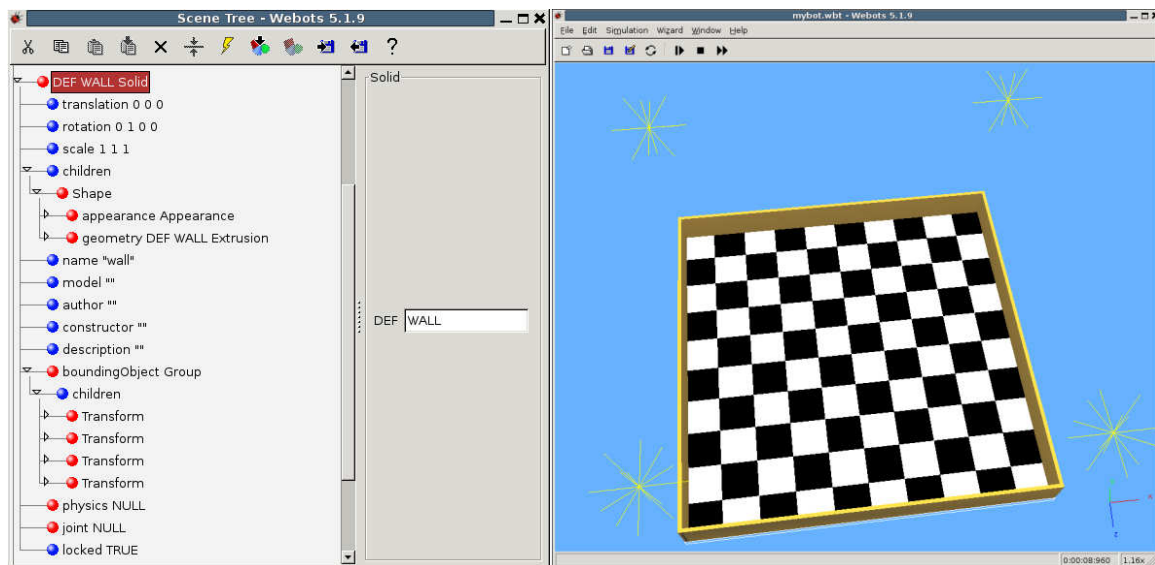
9. В полето „spine” се въвежда, че диапазона на стените по ос „y” е – 0 и 0.1.

10. За предотвратяване преминаването през стените се дефинира boundingObject. В това поле не може да се дефинират комплексни геометрични форми. Обектите които могат да се опишат тук са лимитирани до кутия, цилиндър, сфера, квадрат, правоъгълник. Създаването на стени, които да ограничат работната среда, става чрез описване на четири „кутии”, това става по следния начин: избира се полето „boundingObject” и се създава възел „Group”, който ще съдържа четирите части на ограждащите стени. В този възел се създава подвъзел „Transform” в полето „children”. Добавя се „Shape” в същото „children” поле. Във възела „Appearance” се създава

„Material”. Тъй като сензорите които използва робота са базирани на цветово разграничаване, поставям „diffuseColor” и „specularColor” да бъдат бели. На последно място се създава „Box” в същият „Shape” възел. Размера на тази кутия се задава да бъде – [0.01 0.1 1], задавайки го по този начин размера съвпада с размера на станата. По същият начин, за да може да съвпадне с позицията на станата се поставя полето „translation” на възела „Transform” да бъде [0.495 0.05 0]. И последно, за да може по нататък да се обръщаме към този обект като стена и за да може да се разграничава като обект-стена, се задава име. Името се задава от същият „Transform” възел като се записва „WALL_SHAPE” като „DEF” име.

11. По същият начин се задават и останалите три стени от работната среда. Единственото което се променя е координатите на стените а именно – [-0.495 0.05 0]. Трябва да се въведе и поле „rotation” с координати [0 1 0 1.57].

На фиг. 20 може да се види как точно изглежда работната среда



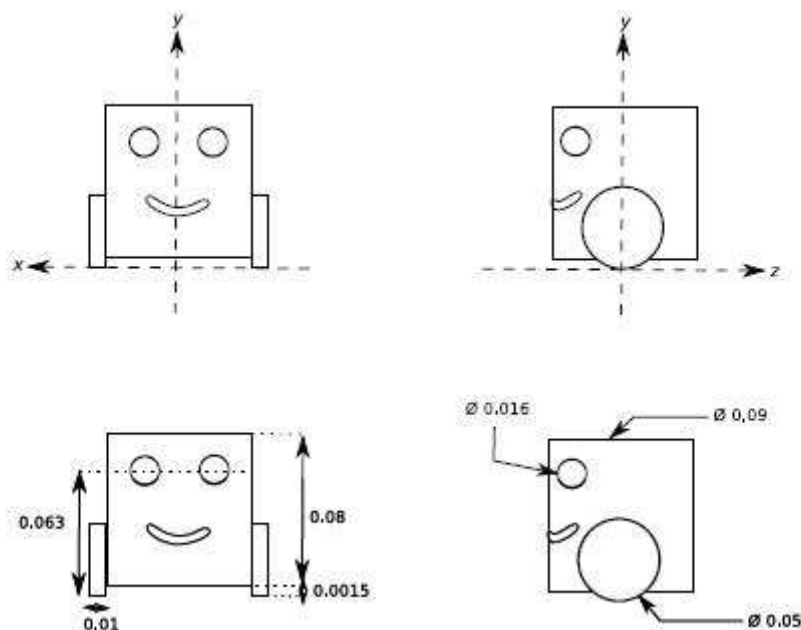
Фигура 20 Работна среда представена във Webots

Имайки средата, трябва да се създадат и препятствия. Това става следвайки следните стъпки:

1. Създава се още един „Solid” възел след последният създаден.
2. По същият начин по който създадох стените на работната среда се създават и препятствията които робота трябва да обходи. В зависимост какви по големина искаме да бъдат тези препятствия се задават различни размери. По подразбиране може да се зададат размери – [0.23 0.1 0.1]. Задава се и име на обекта а именно – „Obstacle”. На последно място се променя полето „translation” да бъде - [-0.05 0.05 -0.25], полето „rotation” не се променя и се оставя такова каквото е по подразбиране.

По този начин се създава препятствие, но в зависимост колко препятствия искаме да има **в** работната среда и с колко робота разполагаме се създават необходимият брой препятствия по същият начин.

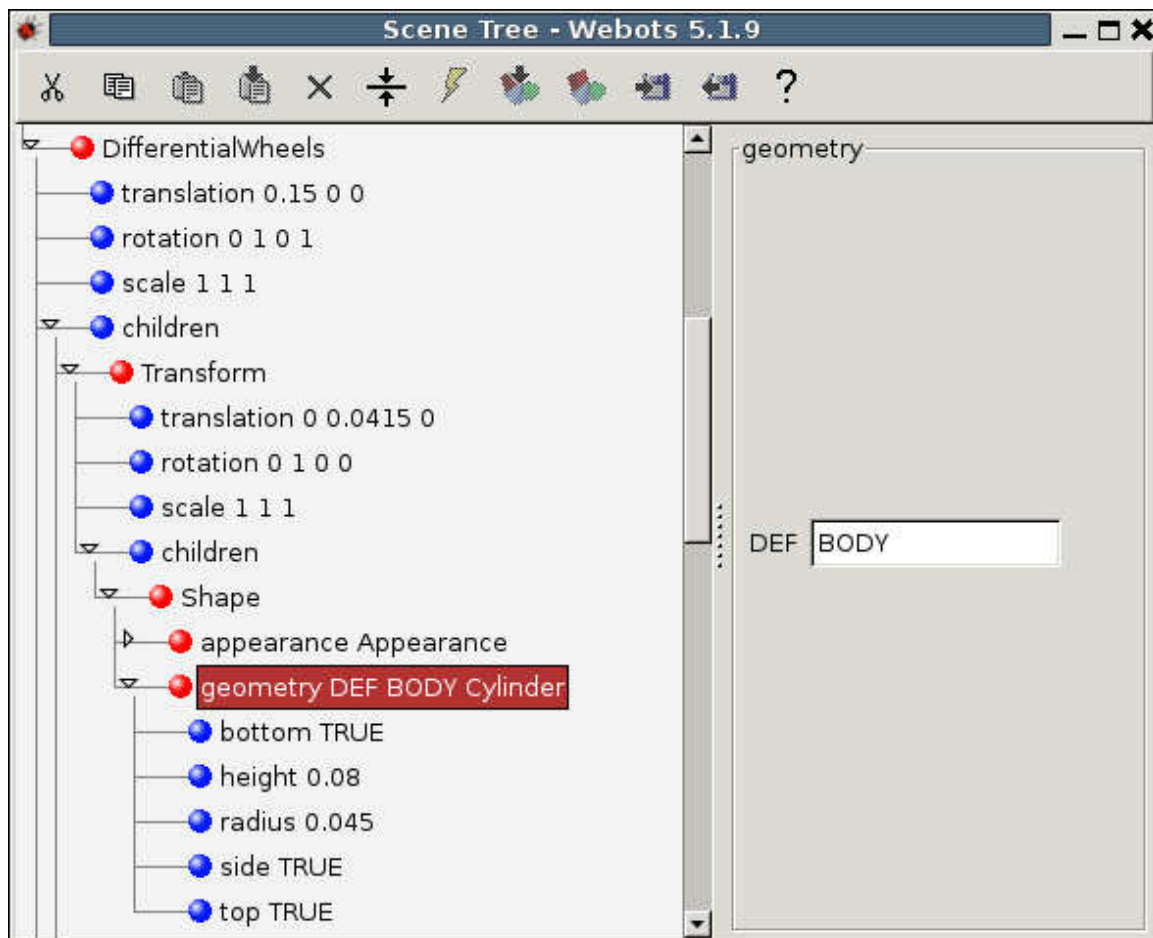
Първоначалните симулации на поставената ни задача, а именно обхождане на среда с препятствия, са направени с обикновен робот показан на фиг .21



Фигура 21 Координатна система и примерен обикновен робот

Тялото на робота се моделира във формата на цилиндър (фиг. 22). Това става по следният начин: Създава се възел „DifferentialWheels”. В този

възел в полето „children” се създава нов възел „Transform”, в който като форма се поставя цилиндър. Този цилиндър има размери съответно – „height” 0.08 и „radius” 0.045. Стойностите на „translation” също трябва да бъдат конфигурирани а именно – [0 0.0415 0].



Фигура 22 Конфигурация на колелата на примерен обикновен робот

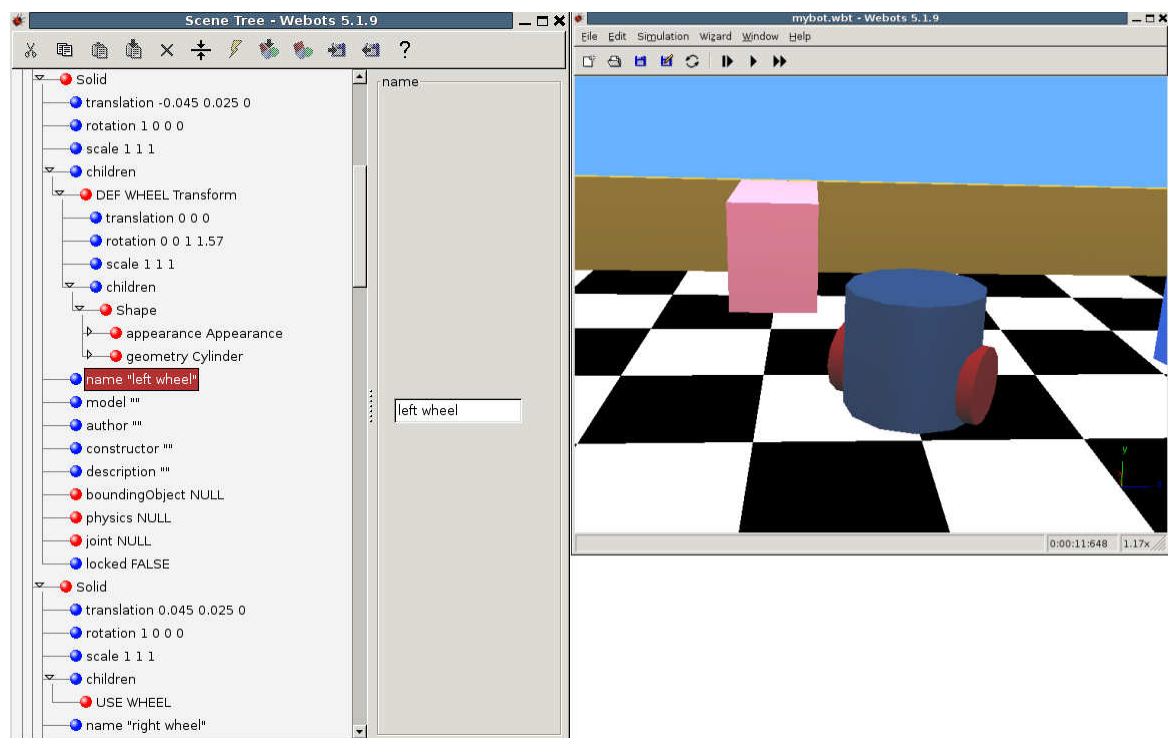
По този начин задаваме ,че робота има формата на цилиндър. Колелата на робота са със същата форма. Лявото и дясното колело трябва да се конфигурират и това става (фиг 23.):

- *ляво колело*

Избира се възел „Transform” който отговаря на тялото на робота и след него се създава нов „Solid” възел, който е за лявото колело. В полето „name” се записва „left wheel”. Оста по която се върти това колело е x . Колелото се върт на $\pi/2$ радиана около оста z . За да не се получава объркване в ротациите на колелото около осите се прибавя възел „Transform” към „children” на възела „Solid”. След като отново се прибави възел „Transform” в полето „geometry” се задава формата на колелто а именно – цилиндър. Размерите на това колело са „height” - 0.01 и „radius” 0.025. Като „rotation” се поставя – $[0 \ 0 \ 1 \ 1.57]$. След това във възела „Solid” се поставя трансляцията на позицията на колелото – $[-0.045 \ 0.025 \ 0]$ и ротацията около оста x да бъде $[1 \ 0 \ 0 \ 0]$.

- дясно колело

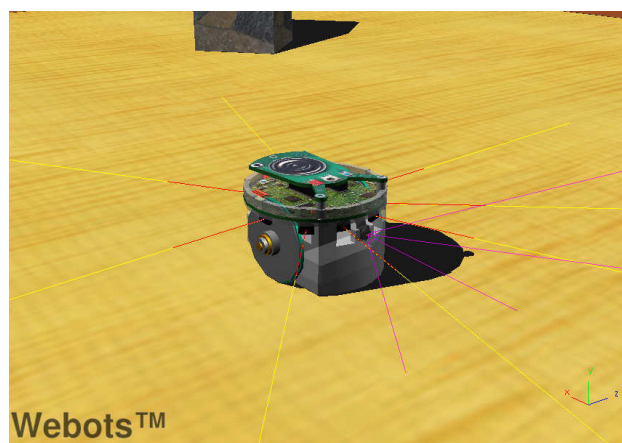
Дясното колело на робота се създава по същият начин както лялото, но с различни трансляции и ротации а именно – трансляция $[0.045 \ 0.025 \ 0]$, ротация – $[1 \ 0 \ 0 \ 0]$.



Фигура 23 Примерен обикновен робот с тяло и колела с форма на цилиндър

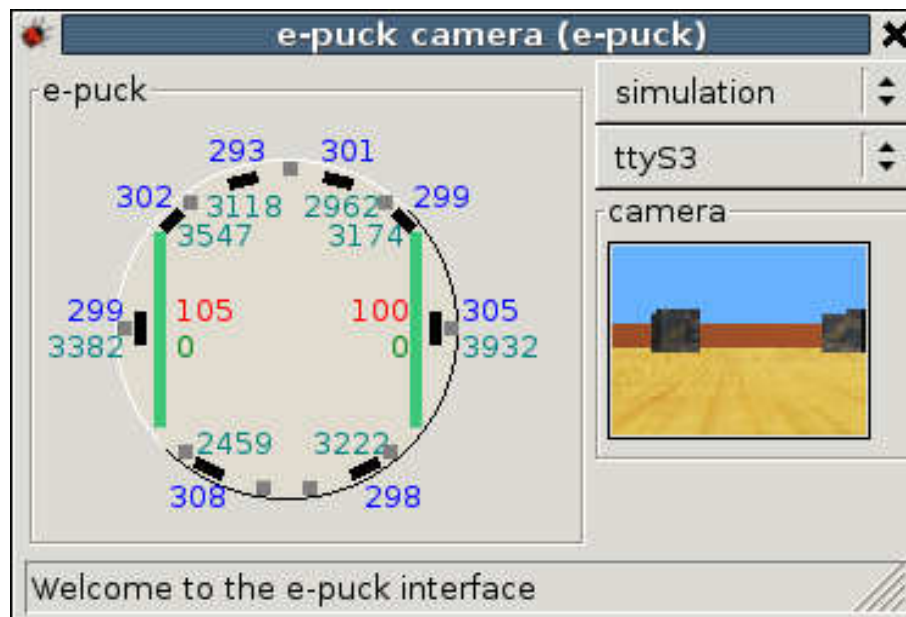
Двата инфрачервени сензори са дефинирани също като цилиндри които са поставени в предната част на тялото на робота. Техният диаметър е 0.016m и са с височина 0.004m. Тези сензори също трябва да бъдат моделирани. Това става като се създава още един възел – „DistanceSensor”.

След направените конфигурации, поставяме робота e-Puck в симулационна среда, фиг 24.



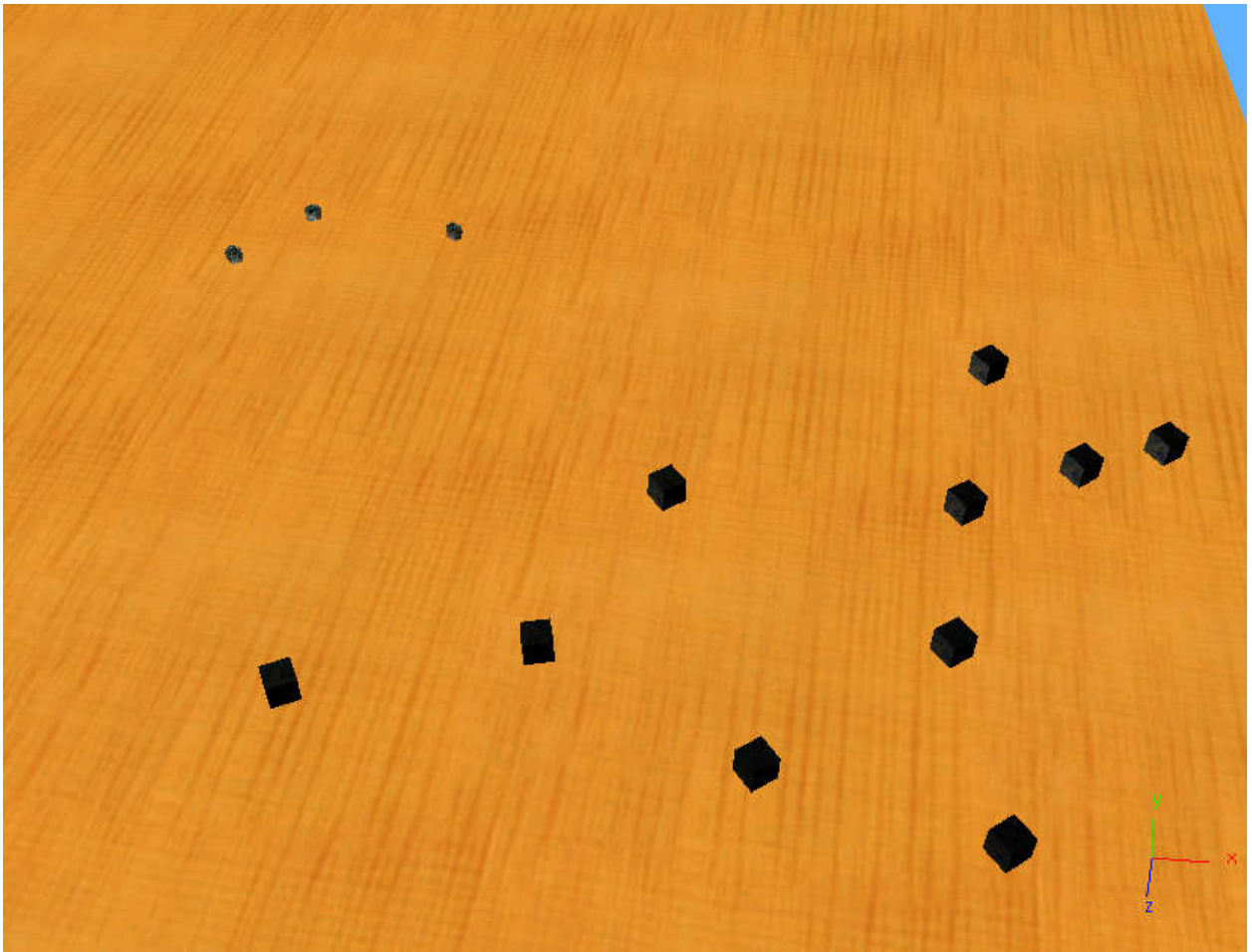
Фигура 24 e-Puck робот в работна среда

След като се настрои и камерата имам абсолютен пълен контрол над робота, фиг 25



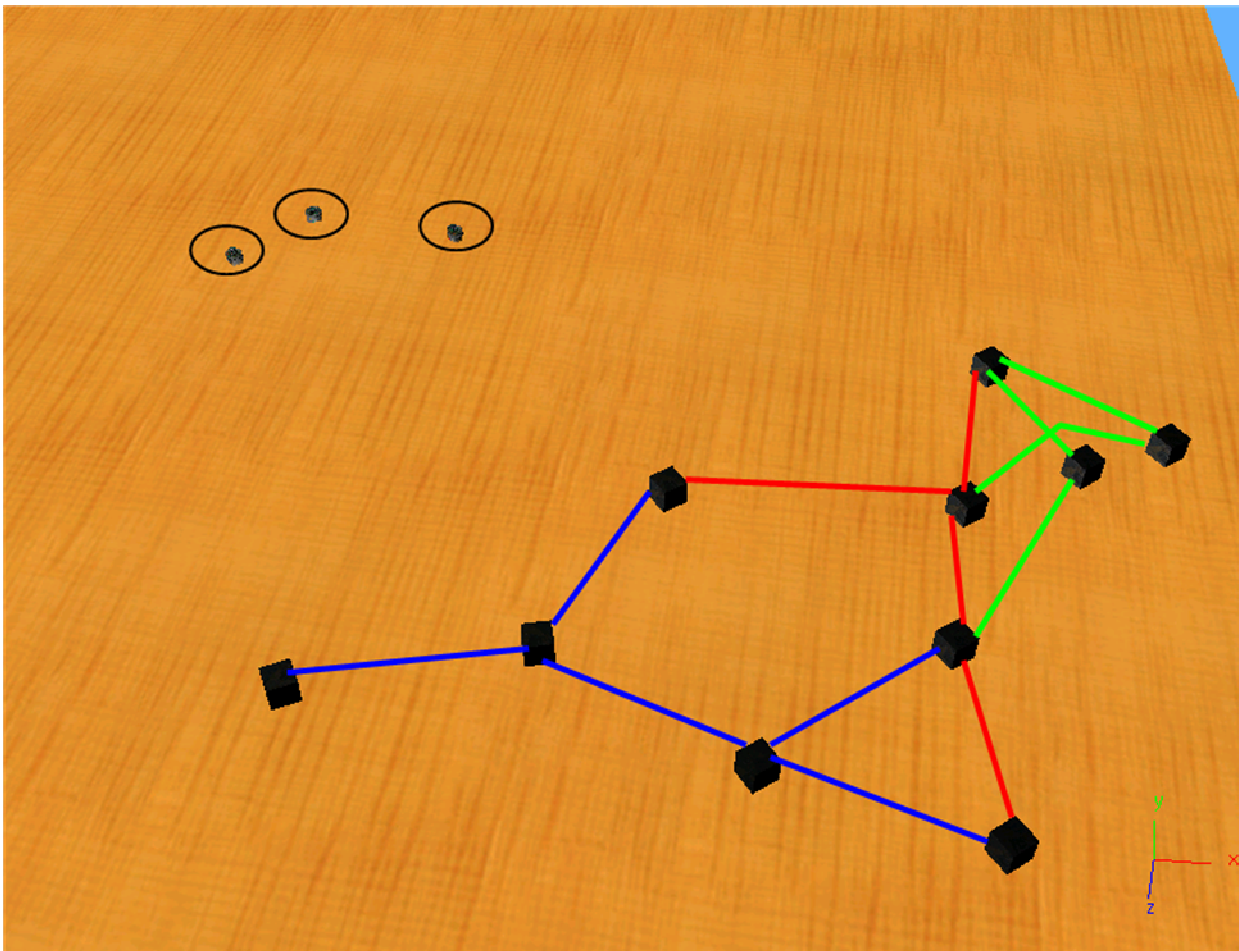
Фигура 25 Прикачена камера на e-Puck робот

Работната среда с препятствията които трябва да се заобиколят от робота е показана на фиг 26.



Фигура 26 Работна среда

Имайки средата прилагам граф визуализация, фиг 27



Фигура 27 Работна среда с вграден граф

3.5 Резултати

Всеки робот има свой локален граф за обхождане. Всяка точка има четири върха, по които да мине робота, за да сме сигурни че пространството ще бъде обходено изцяло. На следващата таблица са показани върховете, линиите и върхове и линии които всеки един робот трябва да обходи.

Робот	Точки	Линии	Върхове и линии за обхождане
Робот 1	6	5	20
Робот 2	7	6	19
Робот 3	8	7	29

Таблица 1 Описание на работи със съответните точки за обхождане

Роботите които са използвани за целите на това задание, са роботи от тип e-Ruck. Използваният симулатор е Webots. На следващата таблица са показани времето и процента на обхождане при три случая:

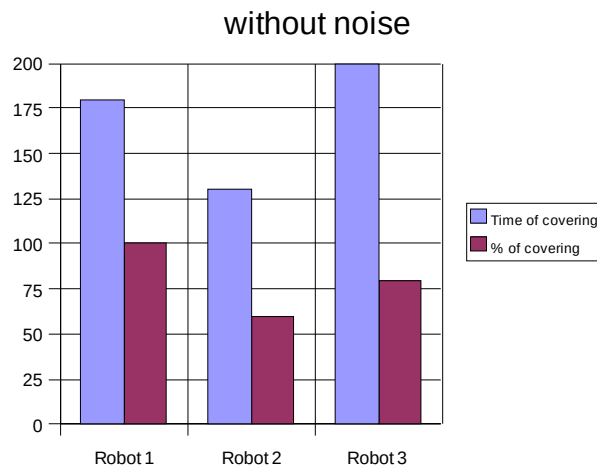
- ако имаме само един робот, който обхожда даденото пространство
- ако имаме два робота, които обхождат даденото пространство
- ако имаме три робота, които обхождат даденото пространство

Всички направени симулации, които са показани като резултати на таблица 2 са направени при идеални условия без наличие на шум.

	Time of covering	% of covering
Robot 1	180	100
Robot 2	130	60
Robot 3	200	80

Таблица 2 Време и процент на обхождане

Таблица 3 Време и процент на обхождане



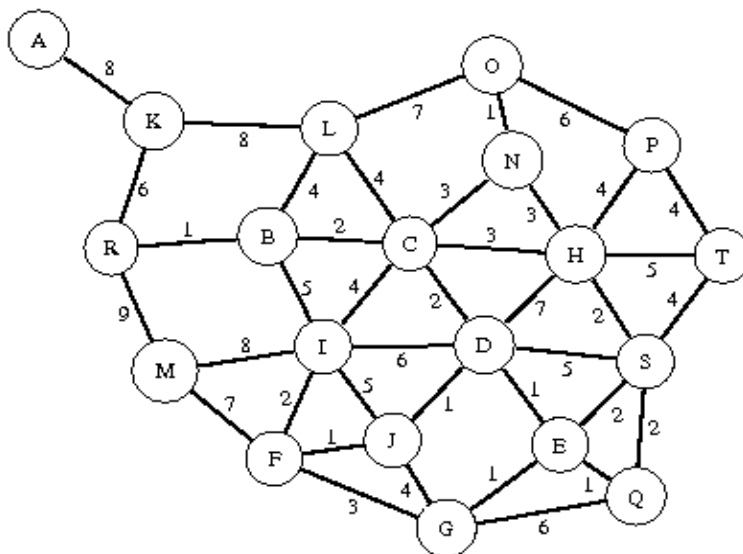
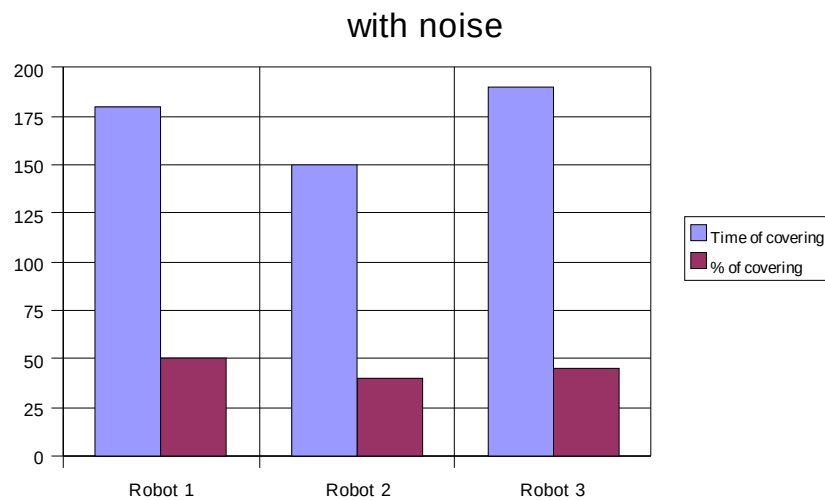
При наличие на шум времето, за което се изпълнява задачата, се увеличава. Понякога дори и само с един робот не се получава цялостно обхождане. На таблица 4. са показани резултати при наличие на шум.

Таблица 4 Време и процент на обхождане

	Time of covering	% of covering
Robot 1	180	50
Robot 2	150	40
Robot 3	190	45

Получените резултати чрез блок диаграми са показани на таблица 5.

Таблица 5 Време и процент на обхождане

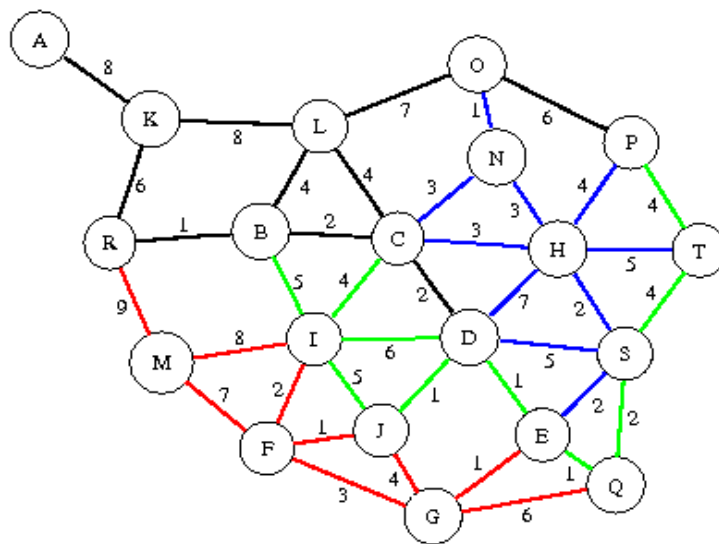


Фигура 28 Разграфяване на граф при наличие на 4 робота

В случай на четири робота имаме общо 20 възела и 38 линии (фиг. 28).

Всеки възел има четири точки, които да бъдат покрити. На фиг. 29. е показан граф, разпределен за четири робота.

Разпределянето на графа за четирите робота е:



Фигура 29 Приложен алгоритъм на Дейстра

В таблица 6 е показано по колко върхове и линии трябва да обходи всеки един робот.

Таблица 6 Разпределение на върхове и линии на обхождане при 4 робота

Робот	Върхове	Линии	Върхове и линии за обхождане
Робот 1	9	10	30
Робот 2	8	9	30
Робот 3	9	10	25
Робот 4	10	10	20

Процентът на покритие на повърхността за всеки един робот е даден в таблица 7

Таблица 7 Време и процент на обхождане

Robot	% of covering
Robot 1	33
Robot 2	50
Robot 3	55
Robot 4	30

Данните от таблица 7 като блок диаграми са показани на таблица 8.

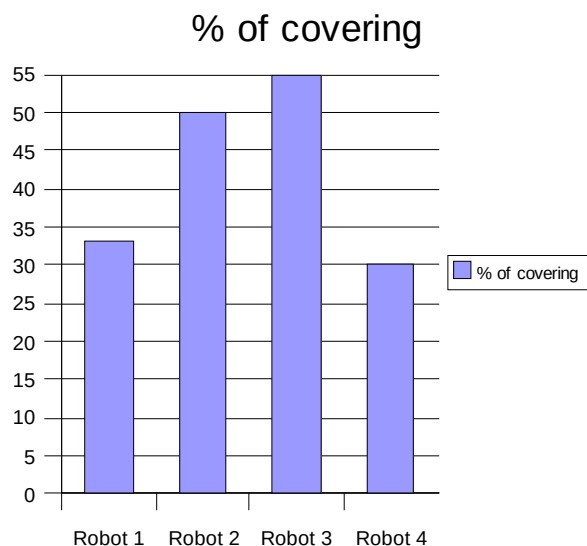
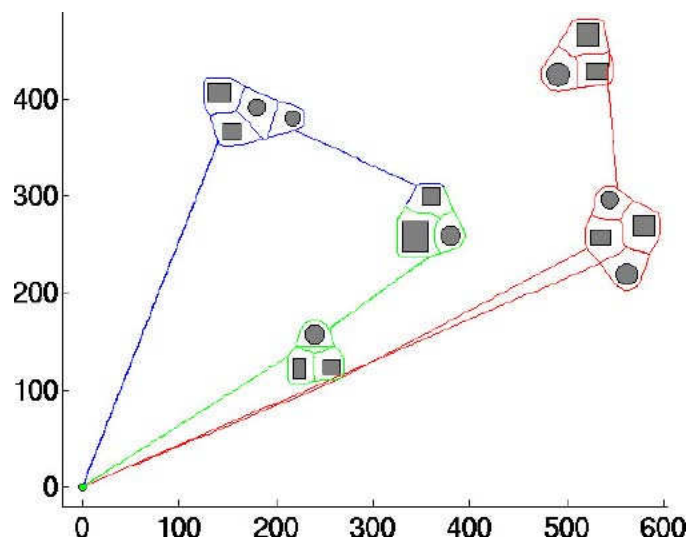


Таблица 8 Време и процент на обхождане

3.5.1 Видове Ситуации

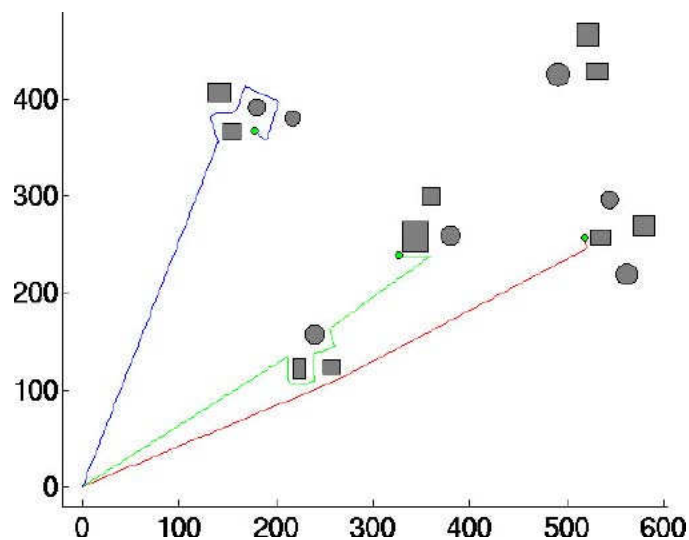
Тук трябва да имаме предвид и ситуация при която робота не може да изпълни зададените му задачи. При такъв сценарий се прилага алгоритъма за проверка. За да се илюстрира подобна ситуация ще разгледаме пример при който разполагаме с три робота и един от тях не може да довърши поставените му задачи.

На фиг. 30 се разглежда случай при нормална ситуация при условие, че и трите робота изпълняват поставените задачи:



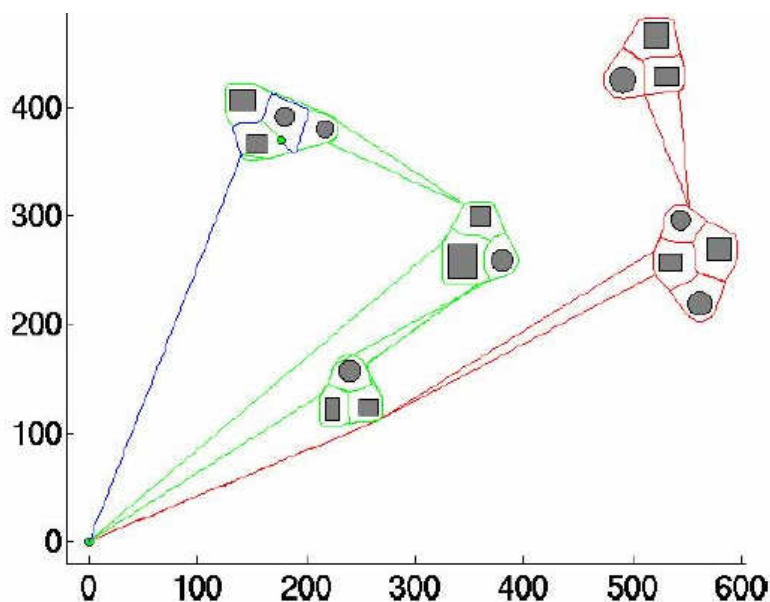
Фигура 30 Среда с три робота, пълно обхождане

Ако предположим че поради възникнали причини единият робот не е в състояние да довърши обхождането на под средата която му е заложена се получава следната ситуация, показана на фиг. 31:



Фигура 31 Среда за обхождане при възникнал проблем при робот със синя линия на обхождане

При ситуация от този род, линиите на обхождане на синия робот се предават на друг робот, в нашия случай се присвояват от робота със зелен контур на обхождане. Разгледаната ситуация може да се види на фиг. 32.



Фигура 32 Ситуация при присвояване на път от страна на друг робот

При подадена грешка от страна на един робот неговата информация се прехвърля на робота с най-близки координати до неговите, поради тази причина робота със зелен контур присвоява маршрута на изпадналия в беда робот.

4 Глава 4 Алгоритми за групово управление на мобилни микро роботи във реално време

Управлението на два или повече робота чрез безжична комуникация е лесна задача, когато роботите не трябва да комуникират помежду си. Задачата, пред която се изправяме е предизвикателството да се направи алгоритъм за синхронно поведение на два робота.

В дадената ни задача, разполагаме с два робота които трябва да пренесат предмет от точка А до точка Б. Трябва да се вземе предвид, че обектът трябва да остане цял и поради тази причина силата, с която роботите захващат предмета, също трябва да бъде контролирана. Предполагаме че за всеки робот се знае силата на захващане. По същият начин предполагаме, че за определяне позицията на всеки робот съществуват данни както за главният командващ компютър, така и за роботите, които са снабдени с адекватна система, създадена за тази цел.

Главната цел на командната система (КС) е да определи тректориите, по които всеки един от роботите трябва да се движи. Друга задача на КС е да предава координатите на всеки един от роботите към другия. Всеки робот е снабден с изчислителни способности.

Комуникационната връзка, по която се предават данни (робот + КС) е безжична комуникация – wireless. От гледна точка на автоматиката и управлението, общият проблем е свързан със закъснението с което се получават и предават данни.

4.1 Безжични комуникационни протоколи

До скоро безжичните комуникации се свързваха само с приложение в мобилните телефони. Но има и други области, където удобството на безжичните комуникации ги е изтласкало на върха на комуникационната вълна. Безжични мрежи, връзка със специализирани сензори, предаване на данни, свързани с експлоатационни параметри и още много са областите в които напоследък безжичните комуникации намират приложение. Използването на радиоканалите за връзка между устройствата, макар че не е ново като изобретение, едва напоследък еволюира значително, особено благодарение на широкото разпространение на Internet, локалните и WAN мрежи, свързващи много хора, позволяващи да си разменят глас, видео и данни помежду си, и то с осезаемо високи скорости. Тласък за развитието именно на безжичните устройства е дала необходимостта от по-голяма свобода и

удобство при изграждането на мрежи, необходимостта от лесното включване на все по-бързо увеличаващия се брой на мобилни абонати, не желаещи да търсят специални точки за включване към мрежата, а глобално погледнато - потребността на съвременния човек от модерни бързодействащи и високоскоростни комуникации. Според изчисленията на аналитиците, през следващата година в света ще има над 1 милиард мобилни устройства, оборудвани с безжична връзка. Поради тази причина и много от фирмите производители съсредоточават своите усилия в тази област и се разработват (напоследък едночипови) устройства, ползващи един или друг протокол за предаване на данни.

Стандарт IEEE 802.11

Комитетът по стандарти на организацията IEEE (Institute of Electrical and Electronics Engineers), IEEE 802, сформира работната група за изработването на стандарт за безжични локални мрежи 802.11 през 1990 г. Задачата на тази работна група е била да разработи всеобщ стандарт за радиопредаване и за безжични локални мрежи, които ще работят на честота 2.4 GHz със скорост на предаване на данни 1 и 2 Mbps (Megabits-per-second) [53].

Работата по създаването на стандарта е завършена през 1997 година, когато през юни месец е ратифицирана първата официална спецификация на 802.11. Стандартът IEEE 802.11 е първият стандарт за безжични мрежи [53] (WLAN, Wireless Local Area Network), приет от независима международна стандартизираща организация, разработила, освен него и множество други спецификации и стандарти за мрежовите връзки по кабел.

Междувременно технологиите за предаване на данни се развиват с нарастващо темпо, така че първоначално заложените в стандарта скорости за предаване на данни от 1 и 2 Mbps вече са малки за големите обеми информация, която се обменят по съществуващите мрежи и са станали безинтересни за потребителите. Това подтиква разработчиците на стандарта IEEE 802.11 към създаване на нови стандарти - IEEE 802.11b, които де-факто се явяват разширения на основния, 802.11, но скоростта на обмен е повишена до 11 Mbps скорост на обмен на данни между безжичните устройства, което в идеалния случай означава над 1.4 MB/сек. трансфер между устройствата.

Стандартът IEEE 802.11 работи в съответствие с двете долни нива на модела OSI (Open System Interconnection) - физическо и канално

ниво [53]. Всяко едно мрежово приложение, протокол или операционна система могат да работят при това положение в една безжична мрежа не по-лошо, отколкото това става в обикновена Ethernet мрежа. Основната архитектура, особености, протоколи и служби са определени в стандарта 802.11, а спецификацията 802.11b засяга физическото ниво, променяйки скоростта на обмен и достъп към по-висока [53].

На физическо ниво са отделени общо три метода за предаване на данни, единият от които е в инфрачервеният диапазон, а другите два са радиочестотни, работещи в интервала между 2.4 GHz и 2.483 GHz. Двата широколентови канала могат да използват различни методи за организиране на предаването - метод на пряка последователност (DSSS-Direct Sequence Spread Spectrum), или метода на честотните подскоци (FHSS - Frequency Hopping Spread Spectrum).

Стандартът 802.11 фиксира два вида безжично мрежово оборудване - *клиент*, ролята на който обикновено се поема от компютър с инсталирана безжична мрежова интерфейсна платка (Network Interface Card, NIC), и *точка за достъп* (Access point, AP), която служи за връзка между безжична и кабелна мрежа.

Клиентът е окомплектован с мрежова карта 802.11, която може да бъде с интерфейс ISA, PCI или PC Card, както и във вид на вградено решение. Точката за достъп обикновено е оборудвана с приемопредавател, интерфейс към кабелна мрежа (802.3) и специализирано програмно осигуряване.

Стандартът IEEE 802.11 определя два режима на работа на безжичната мрежа [53] - режим точка-точка (***Ad-hoc***) и режим клиент/сървър, наричан още режим на инфраструктурата (***infrastructure mode***). Първият режим, точка-точка, наричан още IBSS - *независим набор от служби*, както личи и от заглавието, представлява елементарна като структура мрежа, в която отделните станции се свързват една със друга пряко, без да е необходима точка за достъп. Разбира се, при това положение съществуват някои ограничения от типа на максималния брой устройства, които могат да изграждат такава мрежа, което зависи от типа на безжичното мрежово оборудване и от спецификациите на 802.11.

Режимът клиент/сървър предполага използването на поне една точка за достъп, представляваща специализирано устройство, която да е включена към кабелна Ethernet мрежа, и определен, често ограничен брой крайни безжични работни станции. Този тип конфигурация се

нарича *основен набор от служби* (BSS - Basic Service Set), като при наличието на два или повече BSS се формира *разширен набор от служби* (ESS - Extended Service Set). Очевидно е предимството на режима клиент/сървър, когато безжичната мрежова станция може да получи достъп до локално мрежово устройство или специфична функция, свързано към стационарната мрежа (например към мрежов принтер, скенер или Интернет).

Bluetooth протокол

Консорциумът Bluetooth (IEEE 802.15.1), основан през 1998 г. от IBM, Intel, Toshiba и компаниите за безжични телефонни комуникации Ericsson и Nokia, е разработил стандарта, който създава безжична мрежа, осигуряваща скорост на трансфер 1 Mbit/s [54]. Сега консорциумът включва над 1,500 компании, в това число Microsoft, Dell, Lucent Technologies, Motorola и 3Com. Bluetooth комуникациите се осъществяват в честотния диапазон 2400-2483.5 MHz. Както при IP комуникациите, при Bluetooth данните се изпращат под формата на пакети, контролирани от специални блокове. В зависимост от състоянието на трафика, предаващите и получаващите устройства могат да сменят честотните канали до 1600 пъти в секунда. Освен

адреса на получателя, контролните блокове съдържат и честотата на следващия пакет. Нуждата от постоянен контрол обяснява разликата между теоретичната скорост от 1 Mbit/s и действителната, която обикновено не надвишава 864 Kbit/s [54]. Bluetooth мрежите са проектирани да поемат едновременно до осем устройства. На всяко устройство се присвоява статус slave или master. Master-ите са отговорни за организирането и управлението на комуникациите, включително и тези между slave устройствата. Съществуват и 20 различни профила [55] и три типа връзки, което допринася за високата цена на Bluetooth чиповете. Профилите контролират поведението на Bluetooth устройствата, а връзките определят режимите на предаване между комуниращите модули.

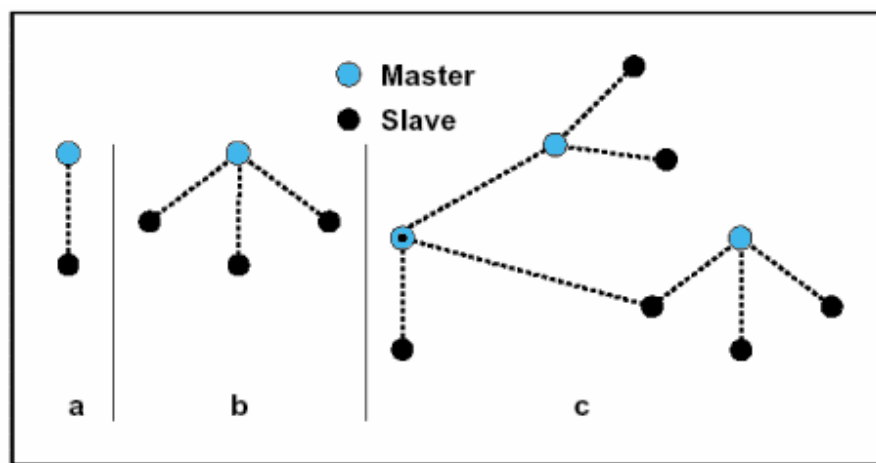
Разстояние, на което могат да се отдалечат две устройства е около 20-30 метра (типичното разстояние обикновено не надвишава 10 метра), но се работи по удължаването му. В замяна на това, няколко Bluetooth устройства могат да се свържат в мрежа и през стена (стени) или на няколко етажа в една сграда, без да има необходимост от пряка видимост или външна антена, по същия начин, по който могат да се свързват IEEE 802.11 Широчината на канала е 723,2 Kb/sec. за устройства, работещи в асинхронен режим, и 433,9 Kb/sec. за

работещите в синхронен режим [9]. Bluetooth има и друга, отличаваща го от останалите технологии особеност: различните Bluetooth устройства влизат в контакт едно с друго автоматично, веднага след като попаднат в обсега на приемо-предавателя, а за установяването на връзката, автентификацията и др. се грижи програмното осигуряване.

Едно от големите предимства на Bluetooth е, че устройството, поддържащо стандарта, влизайки в обхват може да установи връзка не с едно, а с множество други, поддържащи тази технология, като не е задължително те да си взаимодействат активно [9].

Устройство, обменящо активно информация с други устройства, според терминологията на Bluetooth се нарича *master*, а устройствата, с които то комуникира активно се наричат *slave*, като максималния брой *slave* устройства може да бъде 7 [9]. Освен това може да съществуват още неограничен брой неактивни *slave* устройства, които са установили връзка с него, макар, че са синхронизирани с *master*, не обменят данни с последния, очаквайки освобождаване на свободно място, за да осъществят преноса на данни. Такъв тип връзка между устройствата се нарича *piconet* [54, 55]. В рамките на една *piconet* връзка може да има само едно *master* устройство, но когато е необходимо, свързаното с него *slave* може да смени статуса си на *master*, образувайки своя *piconet*

структура. Този тип сложна съставна структура носи наименованието *scatternet* [54, 55], в която всяко едно устройство може да бъде едновременно и master и slave, в зависимост от конкретната ситуация и мястото му в структурата. На фиг. 33 е показан начина на свързване в *piconet* с едно устройство slave (a) и с няколко (b) и свързване от типа *scatternet* (c).



фигура 33 Връзки между Bluetooth устройства в *piconet* с едно устройство slave (a), няколко устройства slave (b) и свързване от типа *scatternet*

Разбира се, за да се избегне дублирането на устройствата и други нежелани отклонения, всяко устройство, освен уникалното си име, взаимодейства с другите, използвайки различен канал за връзка, на различна честота и с различен от другите параметър *hopping* [55], характеризиращ *hopping channel* (хопинг-канал). Хопинг (*hopping*)-това

е периодична промяна на честотата, определяна от параметъра hopping sequence.

Недостатъци [54, 56] на интерфейса Bluetooth са:

- висока (засега) производствена цена на необходимата за функционирането на интерфейса елементна база
- липса на добра поддръжка на ниво операционна система (може да бъде решено с написване на драйвери)
- липса на интерес от страна на производителите да предлагат устройства с Bluetooth интерфейс
- нерешени проблеми със запазването на неприкосновеността на обменяните между устройствата данни (лесно се прихващат и декодират от разстояние)
- ниска скорост на предаване на данни
- малък обхват на устройствата

От друга страна, има предимства, които може би ще помогнат на тази технология да си изгради добро бъдеще (на Bluetooth в сегашния му вид или на негов наследник, изчистен от недостатъците на предшественика).

Основни предимства [54, 56] са:

- добре обмислена структура

- неотклонно намаляваща цена на хардуерния модул (едночипово решение)

- поддръжка от страна на консорциум, основан през 1998 г, със свободно безплатно членство, в който членуват над 2000 компании, между които IBM, Intel, Nokia, Erricson, Toshiba, 3COM, Lucent, Microsoft.

- Bluetooth позволява икономически изгодно, непрекъснато предаване на глас и информация по безжичен път в късовълновия радиодиапазон, което ще даде на потребителите възможност за лесно и бързо свързване на голям брой устройства без необходимост от кабелна мрежа, разширявайки комуникационните възможности на преносимите компютри, мобилни телефони и други приложения вариращи от детски играчки до автоматизирани битови уреди.

Протокол ZigBee

Съществува множество от стандарти, като Bluetooth и WiFi, които са насочени към високоскоростното предаване на данни за глас и звук, безжични мрежи и т.н. До момента, обаче, няма стандарт за безжични мрежи, който да е в полза на специализираните нужди на сензори и управляващи устройства. Сензорите и управляващите устройства не се нуждаят от голяма широчина на предавателния канал,

но имат нужда от латентни и много нискотокови източници, дълъг живот на батериите и батерии за широка гама от устройства. Именно поради тази причина е разработен стандартът IEEE 802.15.4, съвместно със сдружението “The ZigBee Alliance”. Формирано през октомври 2002 г. [57, 58], сдружението е асоциация от компании, работещи съвместно по създаването на евтини, двупосочни безжични стандарти. Прогресивно нараства броят на компаниите, които работят върху решения и продукти, базирани на ZigBee технологията. В момента около 35 компании активно работят върху дефинирането на общ стандарт ZigBee, като Honeywell, Invensys, Mitsubishi, Motorola and Philips (които са петте фирми, основатели на сдружението “The ZigBee Alliance”) и още около 30 компании участнички, производители на безжични устройства [57, 58].

ZigBee технологията е разработена основно за нуждите от евтино решение, базирано на стандарта за безжични мрежи, което да поддържа малки потоци от данни, ниска консумация и надеждност. Това е единствената до момента стандартизирана технология, адресирана към уникалните нужди на мрежови приложения, свързани с дистанционно наблюдение и контрол на различни сензори [59].

ZigBee технологията е подходяща за широк клас от приложения в много области. Естествено, тези приложения, които могат да ползват характеристиките на стандарта IEEE 802.15.4, могат да се ползват от решения, базирани на ZigBee технологията.

Като примери за областите на приложения, могат да се посочат [56, 57]:

- Безжични решения за домашна охрана;
- Дистанционни термостати за климатици;
- Дистанционни осветление и контрол на затъмнеността;
- Бутон за повикване за нуждите на болни и инвалиди;
- Универсални дистанционни за радио и телевизионни приемници;
- Безжични мишки, клавиатури, джойстици;
- Безжични датчици за дим, кислород и др.

Едно от основните предимства на устройствата, ползващи ZigBee технологията е ниската консумация, което от своя страна, дава възможност за многократно увеличаване живота на батериите. Очаква се батериите да удължат живота си до няколко години при използване на стандартни батерии в приложения за наблюдение. Устройствата,

ползващи батерии, като захранващ източник, могат да се установят в неактивен режим (sleep) за минути, дори и за часове [57, 54].

Друга съществена особеност на тази технология, е че ZigBee е в зависимост от това коя честота е селектирана и колко често даденото устройство се нуждае да комуникира. Схемата на адресиране поддържа до 255 възела. При поддържане на 16 канала при 2.4GHz честота, ZigBee мрежата може да съдържа около 4000 отделни възела в единична мрежа с висока надежност на системата [59, 60].

“The ZigBee Alliance” предлага стандартизирана база от решения за сензори и управляващи системи. Целта е да се установи глобален управляващ/сензорен мрежови стандарт.

За разлика от Bluetooth, който има различни режими и състояния, ZigBee/IEEE 802.15.4 има два основни режима – активен (изпраща/приема) и пасивен (“sleep”). Приложният софтуер е необходимо да се фокусира върху самото приложение, а не върху избора на оптимален захранващ режим за всеки различен случай.

Основните технически характеристика на протокол 802.15.4/ZigBee са, [57]:

- Потокът от данни 250Kbs, може да се постигне при 2.4GHz (10 канала), 40Kbs при 915Mhz (16 канала) и 20Kbs при 868Mhz (1 канал);
- Разстоянието за предаване е от 10 до 75 м в зависимост от мощността на захранването и особеностите на терена.;
- Връзката между възлите може да бъде тип “звезда” или “точка до точка” (Peer-to-Peer);
- Много ниска консумация (под 1 mV);
- Поддръжка на устройства с ниска латентност;
- Динамична адресация на устройствата;
- Изключително кратък работен цикъл (под 0,1 %).
- Подобно на WiFi и Bluetooth и стандартът 802.15.4 също работи в долните две нива (физическо и канално) на общия модел за междусистемни връзки (OSI – Open System Interconection).

Стандартът IEEE дефинира два типа устройства. Това са :

- Устройства с пълни функции - Full function device (FFD);
 - могат да работят при всяка топология;
 - могат да работят като координатор на мрежата;
 - могат да бъдат само координатор;

- могат да установяват връзка с всяко от устройствата в мрежата.

- Устройства с редуцирани функции - Reduced function device (RFD);

- могат да работят само при свързване “звезда”;
- не могат да работят като координатор на мрежата;
- могат да установяват връзка само с координатора на мрежата;
- много лесна имплементация.

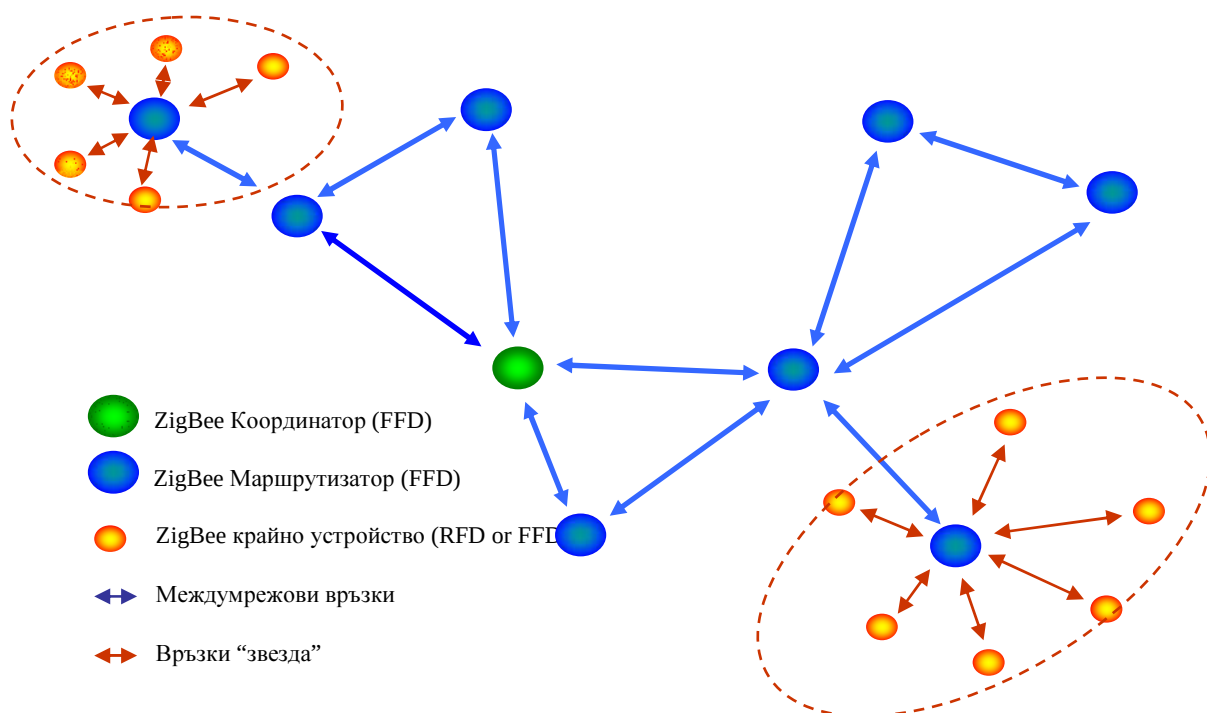
Мрежата IEEE 802.15.4/ZigBee изисква най –малко едно устройство от типа FFD като координатор на мрежата, но крайните устройства могат да бъдат от типа RFD [57, 58].

Основни изисквания към една ZigBee мрежа са :

- Всички устройства трябва да имат 16 битови адреси;
- Кратки (16 – битови) адреси могат да се допускат за намаление размера на пакета
- Методи на адресация
 - за топология “звезда”: адрес на мрежата+ идентификатор на устройство;

■ за топология “peer-peer” – идентификатор на източник/ предназначение.

На фиг. 34. е представен примерен модел на ZigBee мрежа [59].



фигура 34 Връзки между Bluetooth устройства в piconet с едно устройство slave (a), няколко устройства slave (b) и свързване от типа scatternet

Сравнителни характеристики на безжичните протоколи

Сравнение между безжичните протоколи, в буквалния смисъл на думата, не може да бъде направено, но може да се даде класификация, според спецификата на съответния използван протокол. Също така не може да се направи разграничението кой протокол е по-добър, заради различната им насоченост. Може да се направи, обаче паралел, според

който да се определи кой протокол в коя област е най- подходящо да се използва.

Технологии, базирани на протокол ZigBee се използват там, където има мрежи от сензори, контрол на данни и апаратура и където се налага консумацията да е минимална. Тъй като това е най- новия протокол, все още се правят предположения за развитието му и насоките за използване. Много от фирмите предлагат решения за вграждане в бита и развитие на т.нар. “умно жилище”, където интерфейсьт се използва за връзка с всички уреди и до централен команден център.

Bluetooth е протокол, който масово се използва при мобилните телефони, а напоследък и при компютърните системи, когато е необходимо да се премахнат кабелите – безжични клавиатури, мишки и др. Правят се и разработки за вграждането на тази технология в битовата техника.

Протоколът 802.11 или наричан още Wi-Fi основно се прилага за изграждане на безжични компютърни мрежи. Най – простия начин за свързване на два компютъра в безжична мрежа е да се включи във всеки един от тях по една WLAN карта, да се инсталират драйверите и да се настроят необходимите параметри на връзка. Операционните

системи от ново поколение от типа на Windows XP, Linux , след инсталиране на необходимети драйвери разпознават устройствата и позволяват “общуването” с тях.

В таблица 9. е дадена сравнителна характеристика на стандарти и някои безжични протоколи.

Таблица 9. Сравнителна характеристика на протоколи

Протокол > Характеристика	ZigBee™ 802.15.4	Bluetooth™ 802.15.1	Wi-Fi™ 802.11b
Приложение	Наблюдение и контрол	Заместване на кабелите	Web, Video, Email
Системни ресурси	4KB-32KB	250KB	1MB
Живот на батерията (дни)	100-1000+	1-7	1-5
Възли в мрежа	255/65K+	7	30
Честотна лента(kbps)	20-250	720	11,000+
Обхват (метри)	1-75+	1-10+	1-100
Ключови характеристики	Стабилност, Ниска консумация, Евтино	Цена, Удобство	Скорост, Гъвкавост

4.2 Проблеми възникващи при работа в реално време

Зрението е най-важния компонент, помагаш в природата на организмите да съгласуват движенията си с тези на съседите им. Например ято птици и пасажи риби могат да поддържат съгласувано разположение без явна комуникация между индивидите. Дългосрочна задача в този аспект е тази, която да позволява на група мобилни роботи да поддържат визуално дадено разположение в пространството в отсъствието на комуникация. За да се постигне това, се разглежда сценарий, при който следящият робот използва сегментация (разделяне) на движението, за да оцени позицията на другия робот в групата и 3D зрение за следене и избягване на препятствия.

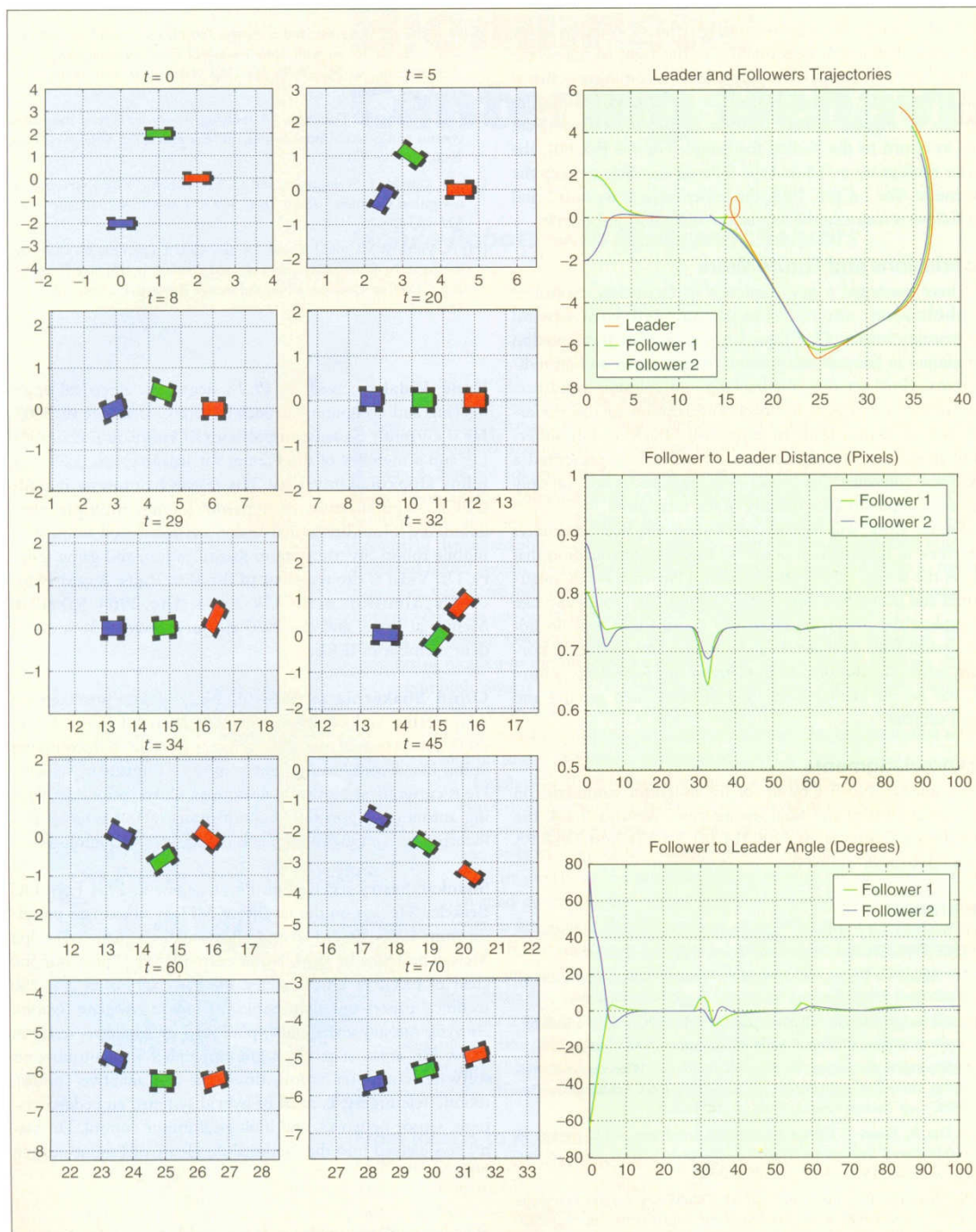
При предходни изследвания, посветени на съгласуващо управление на група роботи се е допускало, че комуникацията между роботите е възможна и вниманието е било концентрирано върху устойчивостта и синтеза на контролер. Предложено е понятието “линия на устойчивост” и са изведени достатъчни условия за устойчивост при разположение по тази линия. Устойчивостта на разположението е представена чрез произволни графи и е предложен критерий за

устойчивост, подобен на този на Найкуист, който може да се изведе на базата на спектралните свойства на графа на Лаплас. Изведени са децентрализирани закони за управление, които водят до устойчиво разположение, при условие че желаната скорост на работа водач (лидер) е известна.

При отсъствието на комуникация между роботите съгласуващото управление е доста предизвикателно от гледна точка на сензорика заради необходимостта от симултантна оценка на движението на група от подвижни обекти. Един от подходите за справяне с трудностите в сензориката е да се боядиса всеки лидер в различен цвят и после да се използва отслеждане по цвят за откриване и следене на лидера.

В [35][36][37][38] е представен нов подход за съгласуващо (синхронизиращо) управление на базата на зрителна сензорна система, при нехолономни роботи, снабдени с централни панорамни камери, за които откриването и следенето на лидерите се базира единствено на тяхното движение в равнината на образа. Подходът се основава на това, да се трансформира задачата за съгласуващо управление от конфигурационното пространство в задача за управление на всеки следящ робот на базата на информация от зрителни сензори. Показано е как да се оцени позицията на всеки лидер в равнината на образа на

следащия робот, използвайки ограничение за ранга на централния панорамен оптичен поток през множество кадри, фиг.35



Фигура 35 Подходи

4.3 Опростен модел на управление на два мобилни микро робота

Двата робота зависят един от друг и поради тази причина моделът който е направен е следният:

$$M_1 \ddot{x}_1 + F \dot{x}_1 + K(x_1 - x_2) = u_1 \quad (1)$$

$$M_2 \ddot{x}_2 + F \dot{x}_2 + K(x_2 - x_1) = u_2 \quad (2)$$

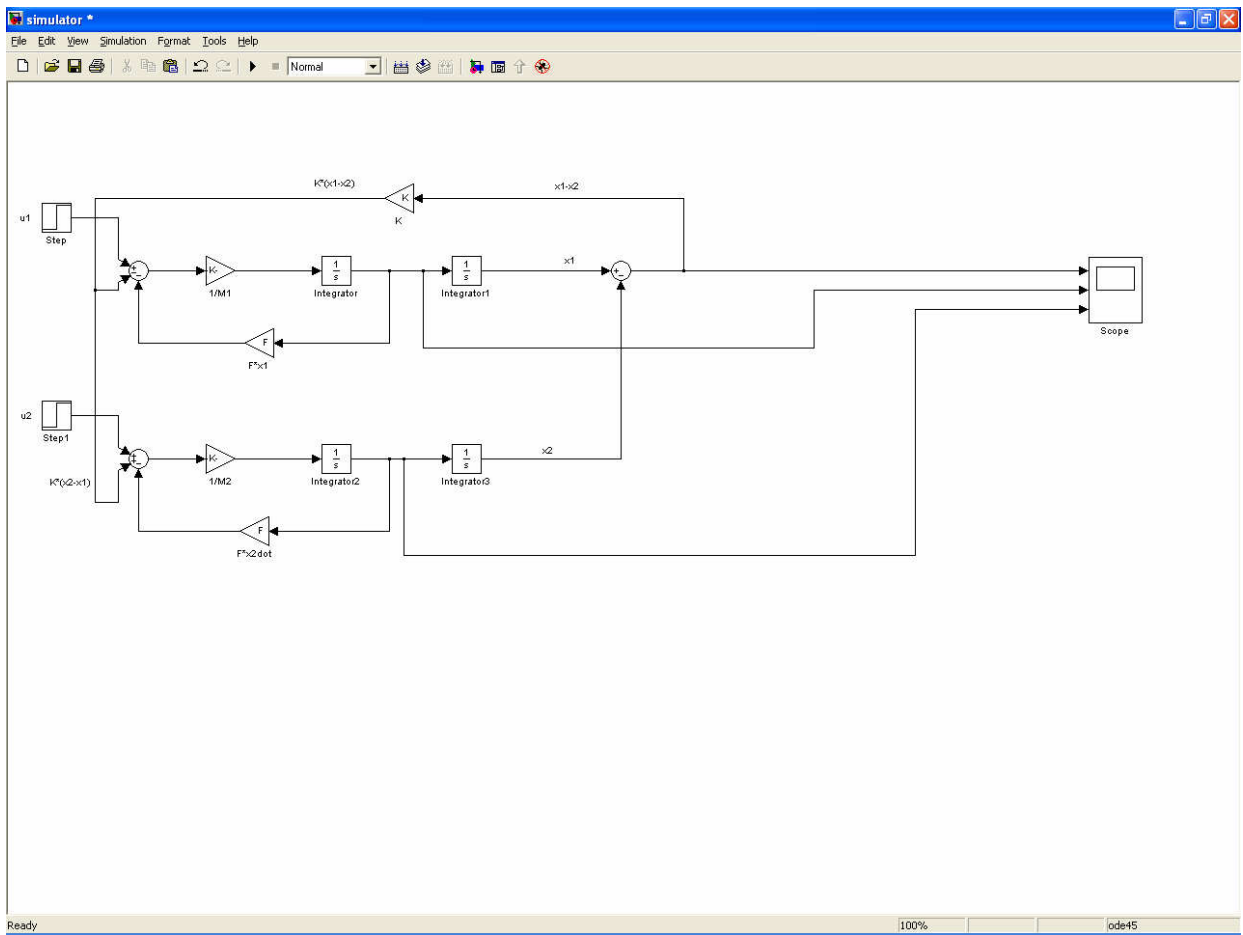
Където M е масата на роботите, F е триенето, K е съгласуването между двата робота.

Лаплас трансформация на горните уравнения ни дава системата, по която се работи по нататък за решаване на проблема.

$$A = \begin{bmatrix} 0 & 1 & 0 & 0; \\ -K/M_1 & -F/M_1 & K/M_1 & 0; \\ 0 & 0 & 0 & 1; \\ K/M_2 & 0 & -K/M_2 & -F/M_2 \end{bmatrix}; \quad (3)$$

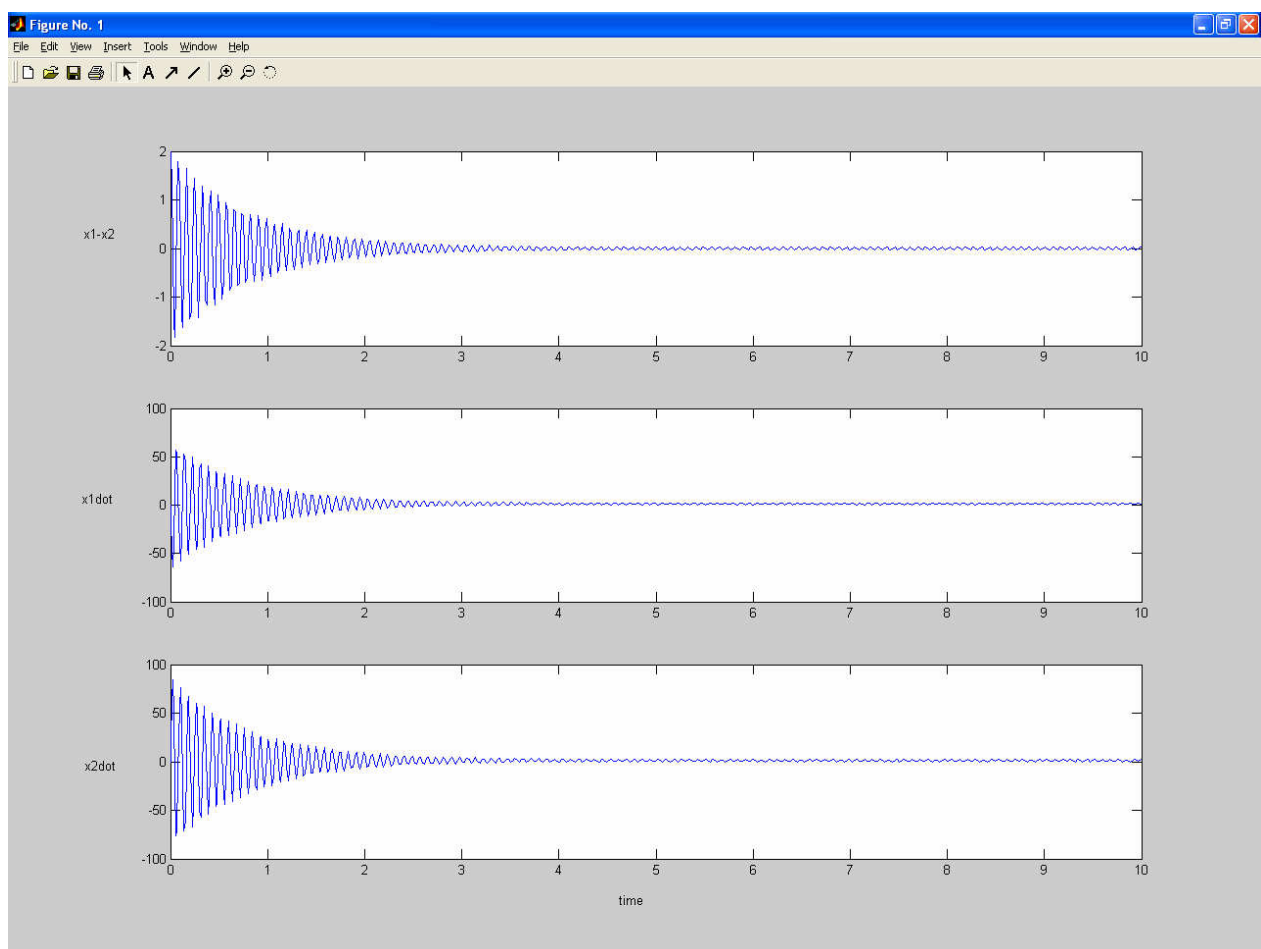
$$B = \begin{bmatrix} 0 & 0; \\ 1/M1 & 0; \\ 0 & 0; \\ 0 & 1/M2 \end{bmatrix}; \quad (4)$$

За целите на моделирането е използван Matlab/ Simulink програма за симулиране. На следващата фигура е показана схемата, съставена от горните уравнения. Фигурата показва два робота, свързани по начин така, че вторият робот следва първия, фиг. 36.



Фигура 36 Опростена блокова схема

Показаната по-горе блокова схема изразява опростен модел на поставената задача пред която сме изправени, за да намерим алгоритми за контролиране. Този модел се отнася за абсолютно идеални условия без шум, без смущения и без закъснения. Сигналите, които се получават са следните, фиг. 37:



Фигура 37 Изходен сигнал

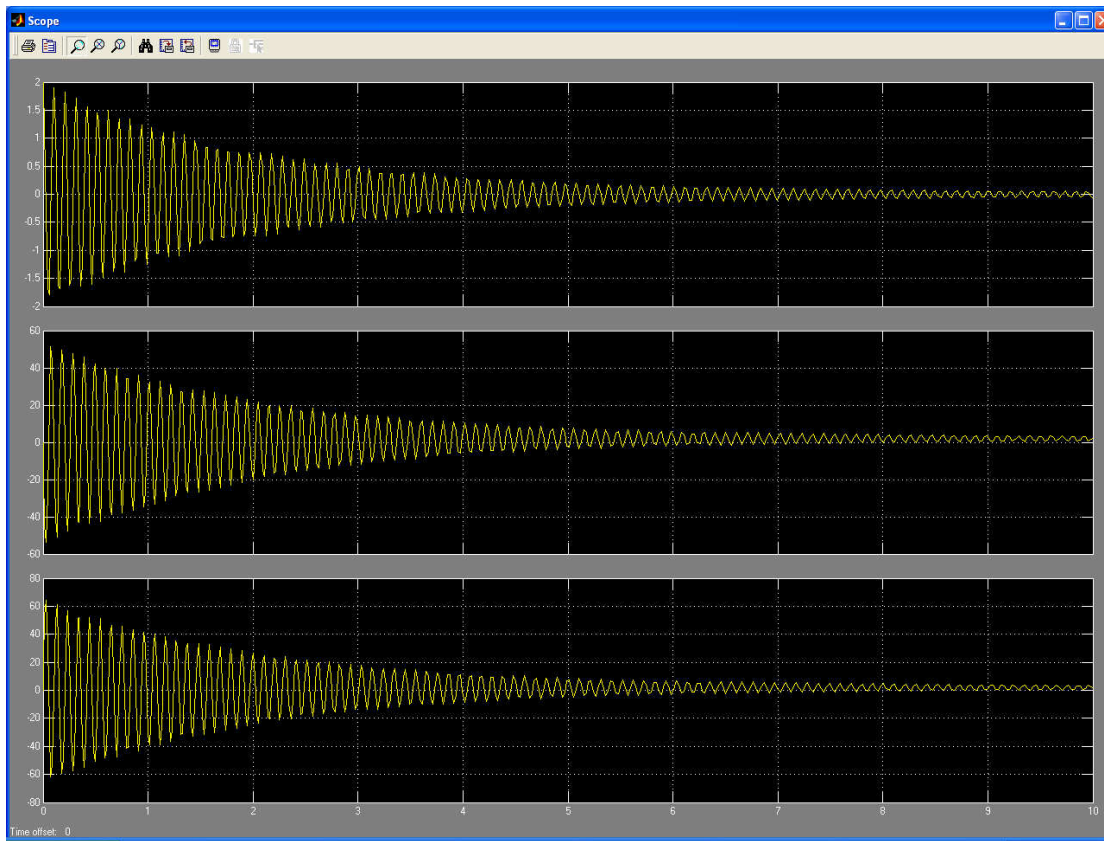
Времето за което протича една симулация е 10сек. От фиг 37. се вижда че при този вид конструкция моделът не е устойчив в началото, но с напредване на симулацията моделът става устойчив и сигналът отива към устойчиво състояние. Триенето F , което се използва в този вариант на модела е:

$$F = 8 \cdot (1e - 1) \quad (5)$$

Съответната константа:

$$K = 1e2 \quad (6)$$

В зависимост от това каква стойност е избрана за F (триене) сигналът затихва по-бързо или по-бавно. След няколко изследвания се стигна до заключението че при по-големи стойности на F по-бързо се стига до устойчиво състояние. На фиг 38. е показано, че наистина при по-малки стойности на F се изисква повече време да се обработят сигналите и да се предадат до всеки един от роботите.



Фигура 348 Изходен сигнал при различни стойности

Разбира се масата на всеки един робот също има своето значение, както и параметъра K . Стойността за масата на всеки един от роботите не би могла да се промени, но за постигане на добри резултати може да променяме стойностите на параметрите K и F .

На фиг 30. са взети под внимание следните стойности на масата на роботите, както и за триенето и константата за съгласуване между двата робота:

$$M_1 = 0.6 \quad (7)$$

$$M_2 = 0.5 \quad (8)$$

$$F = 5 \cdot (1E-1) \quad (9)$$

$$K = 1E3 \quad (10)$$

Регулаторът, който е използван за управление, е LQR регулатор (Linear – Quadratic state feedback Regulator) [39][40]. Този вид регулатор се използва когато трябва да се оптимизира контролера на дадена система. За системи, които не са постоянни, този вид регулатор се задава със следното уравнение:

$$\dot{x} = Ax + Bu \tag{11}$$

Контролерът се оптимизира като се минимизира индекса J , който е зададен със следното уравнение:

$$J = \int_0^{\infty} (x^T Q x + u^T R u) dt$$

Матриците Q и R се коригират в зависимост от нуждите на контролера.

LQR е математичен метод с точно определени уравнения. Когато индекса J е опростен, контролерът който зависи от R, не може да бъде подобрен по-нататък. Главното предимство на LQR контролера е че предоставя точна методология за това как да се оптимизира регулатора.

В нашия случай начинът, по който са пресметнати стойностите за матриците R и Q са следните:

$$Q = c' * \text{diag}[0.001 \ 1] * c \quad (12)$$

$$R = \begin{bmatrix} 2 & 2 \\ 2 & 2 \end{bmatrix} \quad (13)$$

Където c е матрица от вида

$$c = \begin{bmatrix} 1 & 0 & 0 & 0 \\ K & 0 & -K & 0 \end{bmatrix} \quad (14)$$

След стартиране на алгоритъма се получават следните резултати за всяка една от матриците:

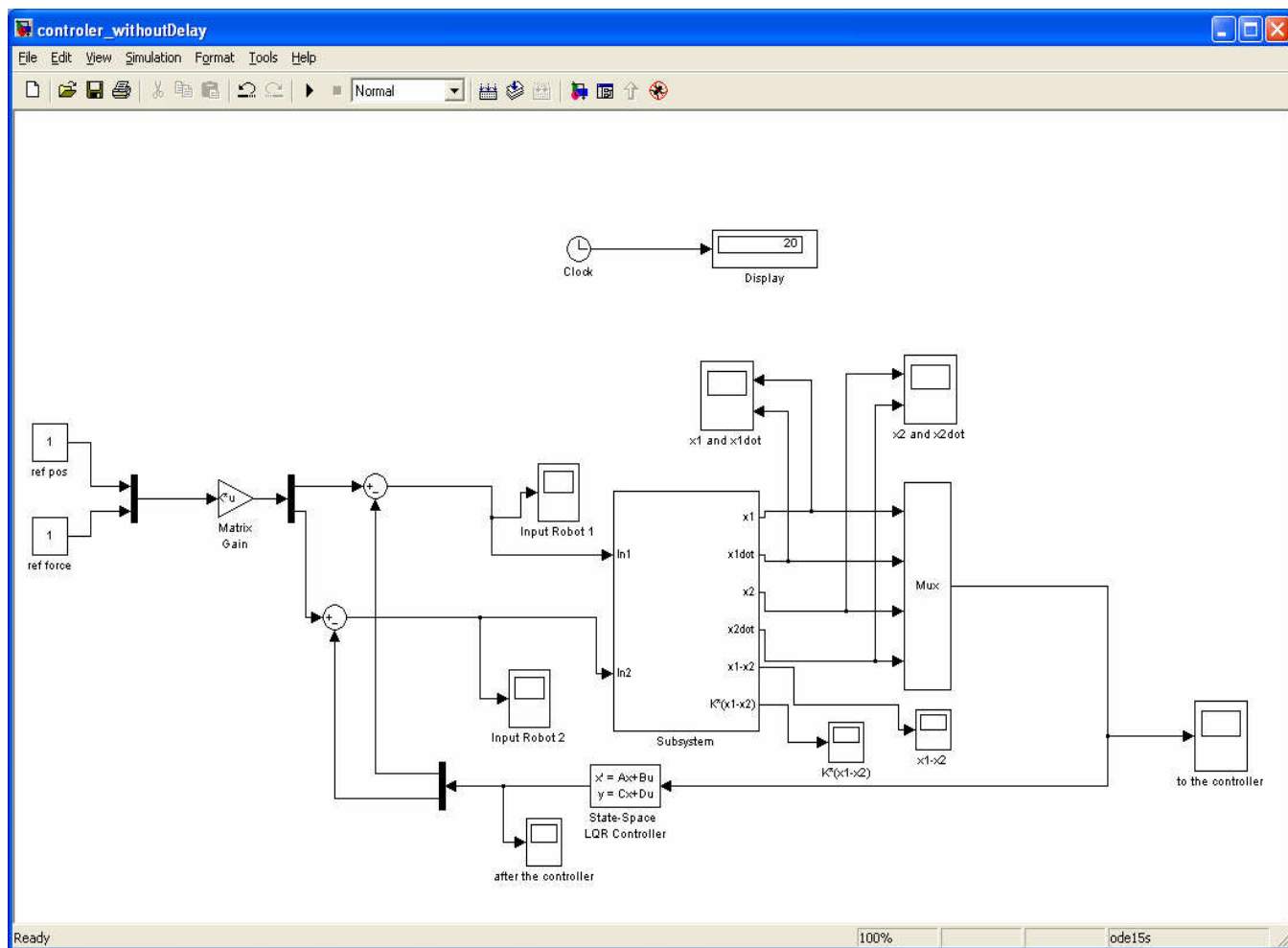
$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -K/M_1 & -F/M_1 & K/M_1 & 0 \\ 0 & 0 & 0 & 1 \\ K/M_2 & 0 & -K/M_2 & -F/M_2 \end{bmatrix} \quad (15)$$

$$B = \begin{bmatrix} 0 & 0 \\ 1/M_1 & 0 \\ 0 & 0 \\ 0 & 1/M_2 \end{bmatrix} \quad (16)$$

$$c = \begin{bmatrix} 1 & 0 & 0 & 0 \\ K & 0 & -K & 0 \end{bmatrix} \quad (17)$$

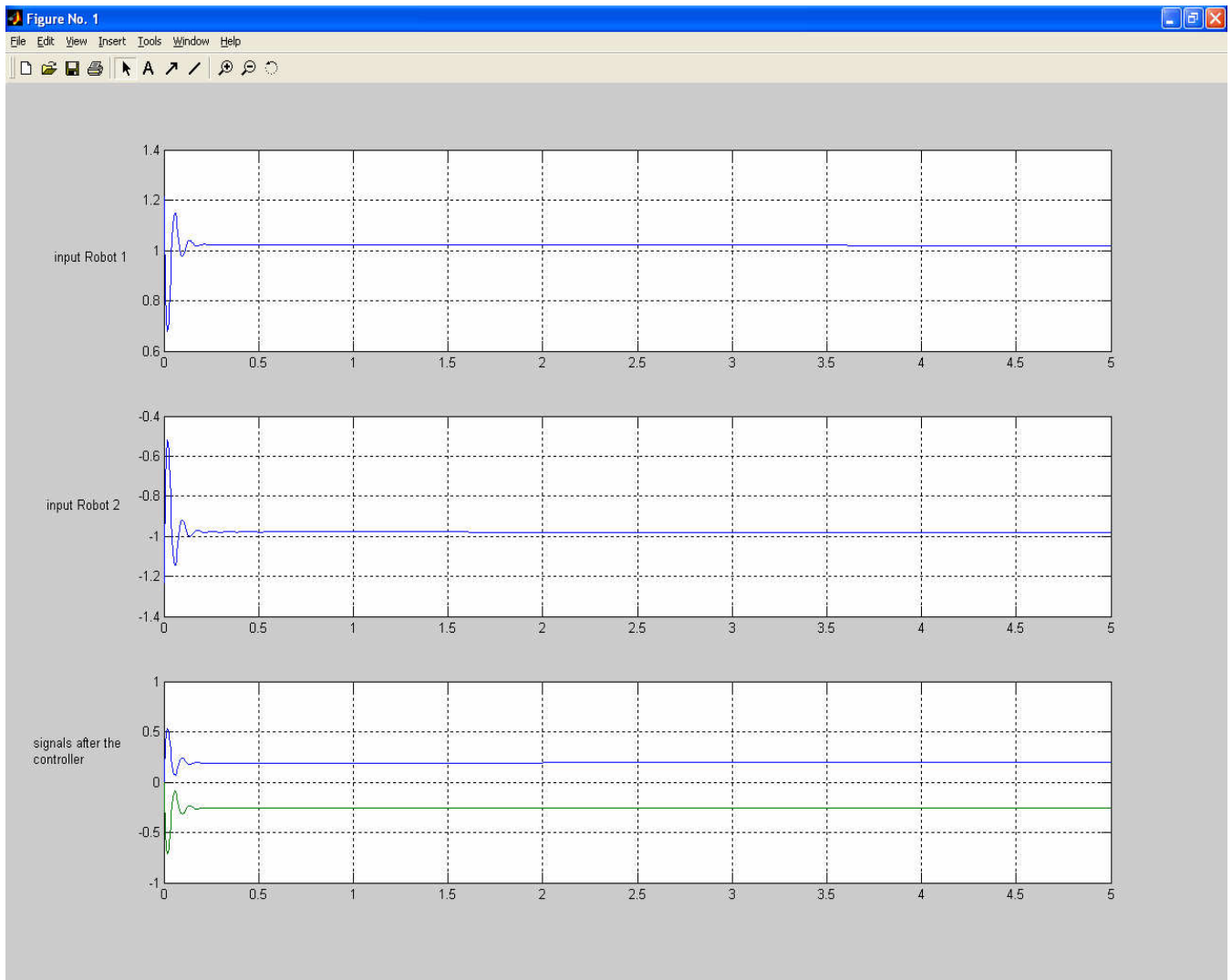
$$d = \begin{bmatrix} u_1 & u_2 \\ y_1 & 0 & 0 \\ y_2 & 0 & 0 \end{bmatrix} \quad (18)$$

След като е построен контролера и са изчислени всички му параметри, той трябва да се включи към симулатора. На следващата фиг. 39 е показан симулаторът с включен контролер



Фигура 39 Симулация с два робота при включен контролер

Съответните входни и изходни сигнали на всеки един от роботите, са показани на фиг. 40

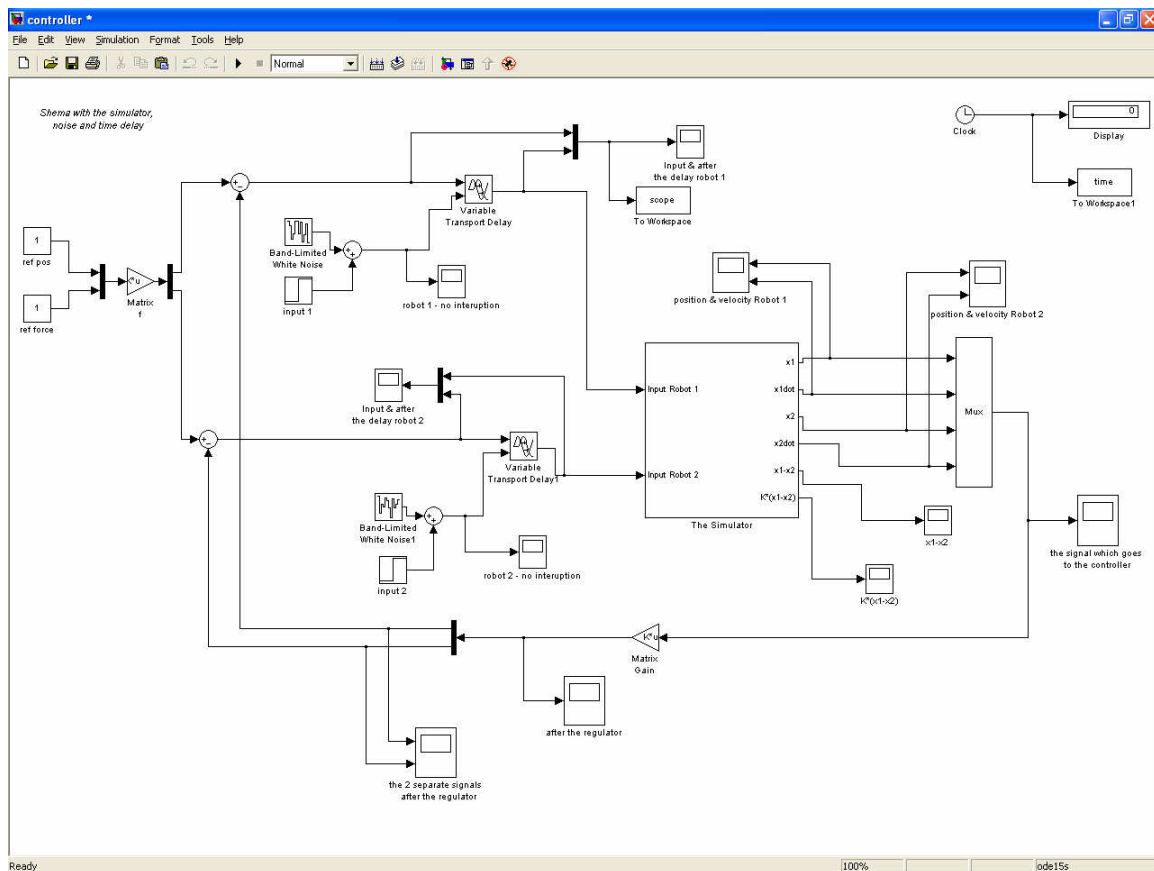


Фигура 40 Изходни сигнали

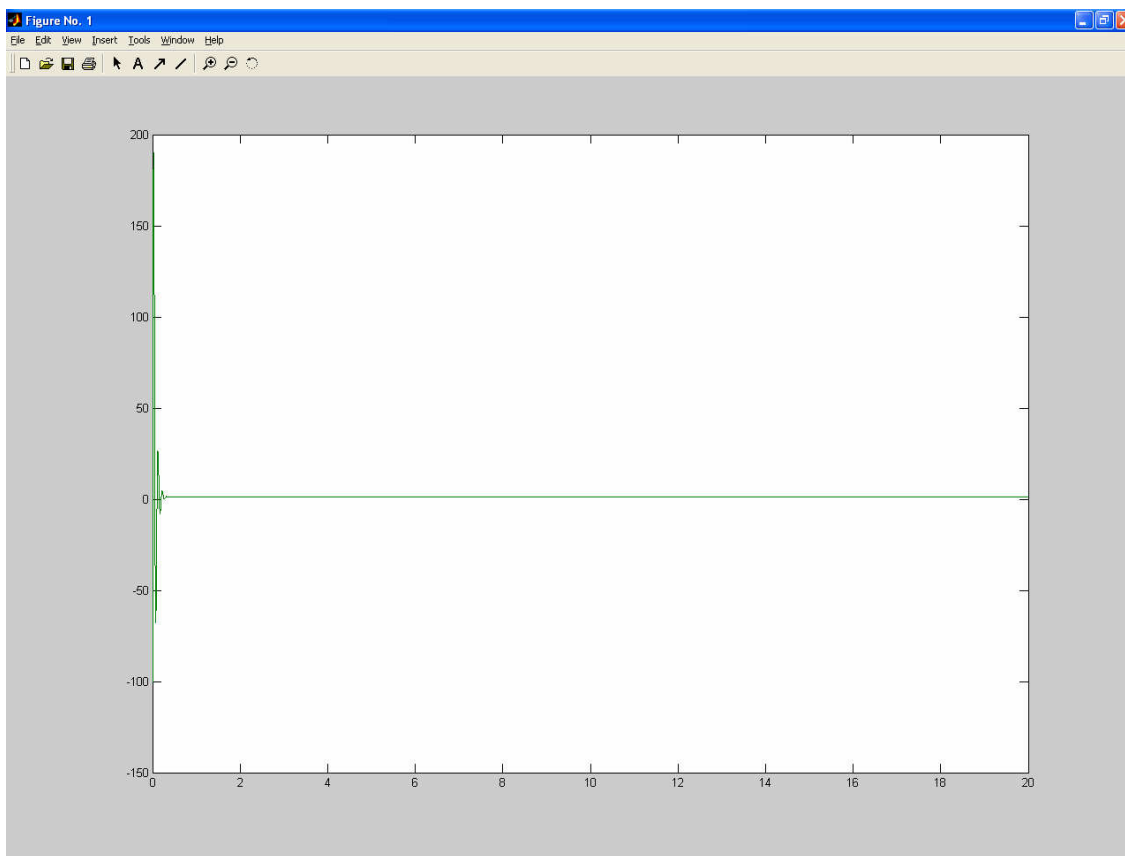
Както вече беше споменато, при предаване на данни с безжична комуникация се получава известно закъснение при получаване и изпращане на сигналите. Това закъснение е от няколко милисекунди, но въпреки това трябва да се вземе под внимание.

4.4 Модел на комуникация на два робота при наличие на шум

Трябва да се вземе предвид че при нормални условия има смущения които представляват проблем понякога при предаване на сигнали и комуникация между роботите и командният център. Поради тази причина се симулира бял гаусов шум. На фиг. 41 е показан модел, при който е включен и бял гаусов шум

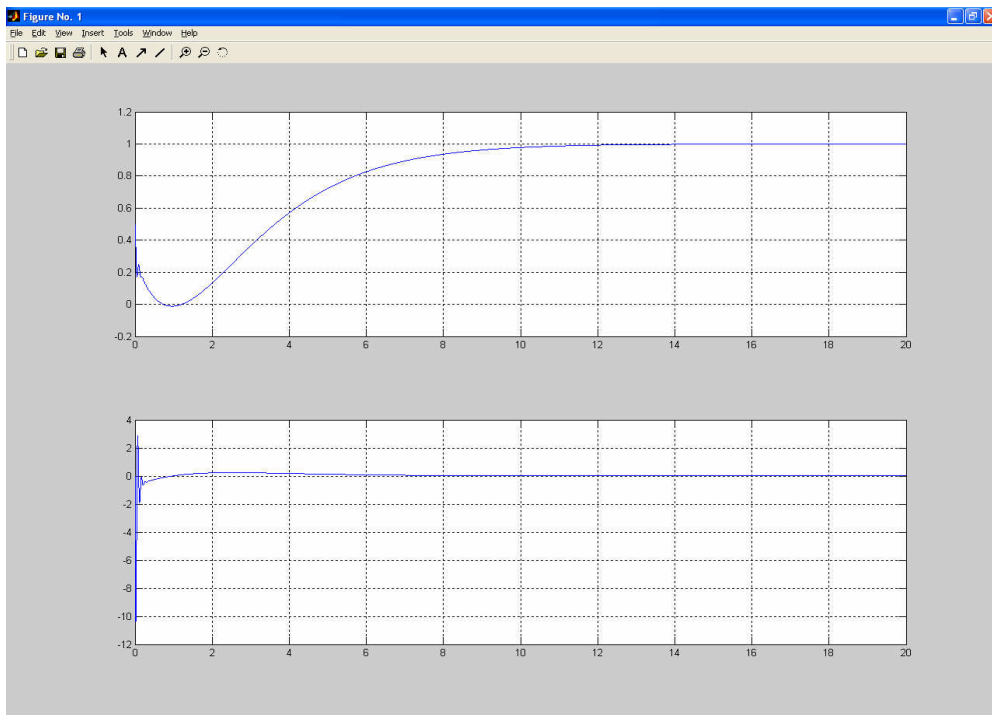


Фигура 1 Симулация при наличие на бял гаусов шум

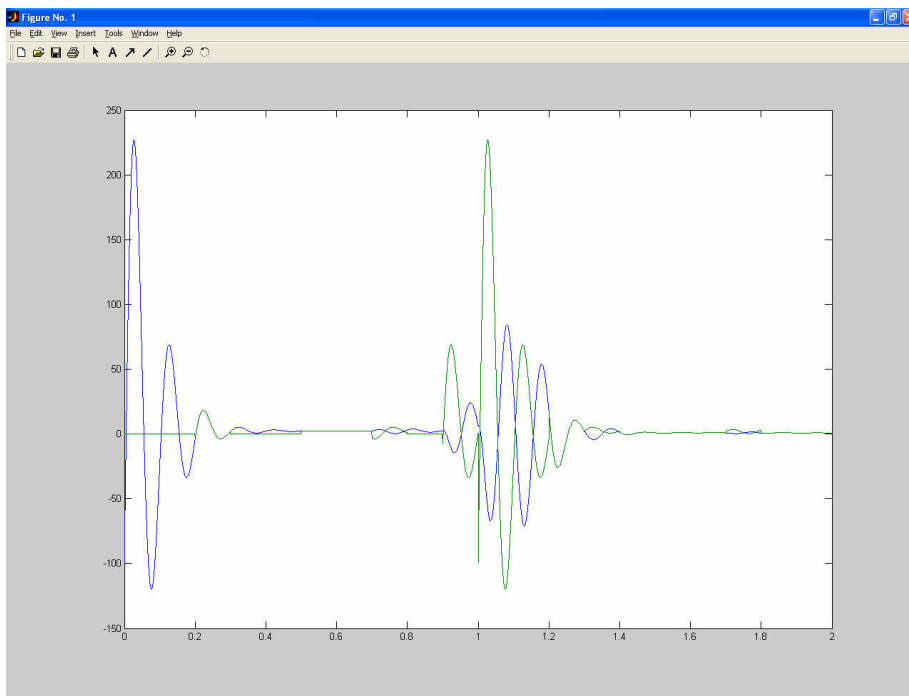


Фигура 42 Изходен сигнал

По същият начин са показани сигналите на робот 1 от същата схема, изразяващи позицията и скоростта на робота, съответните сигнали могат да се разгледат на фиг. 42, 43, 44.



Фигура 43 Изходен сигнал на скорост

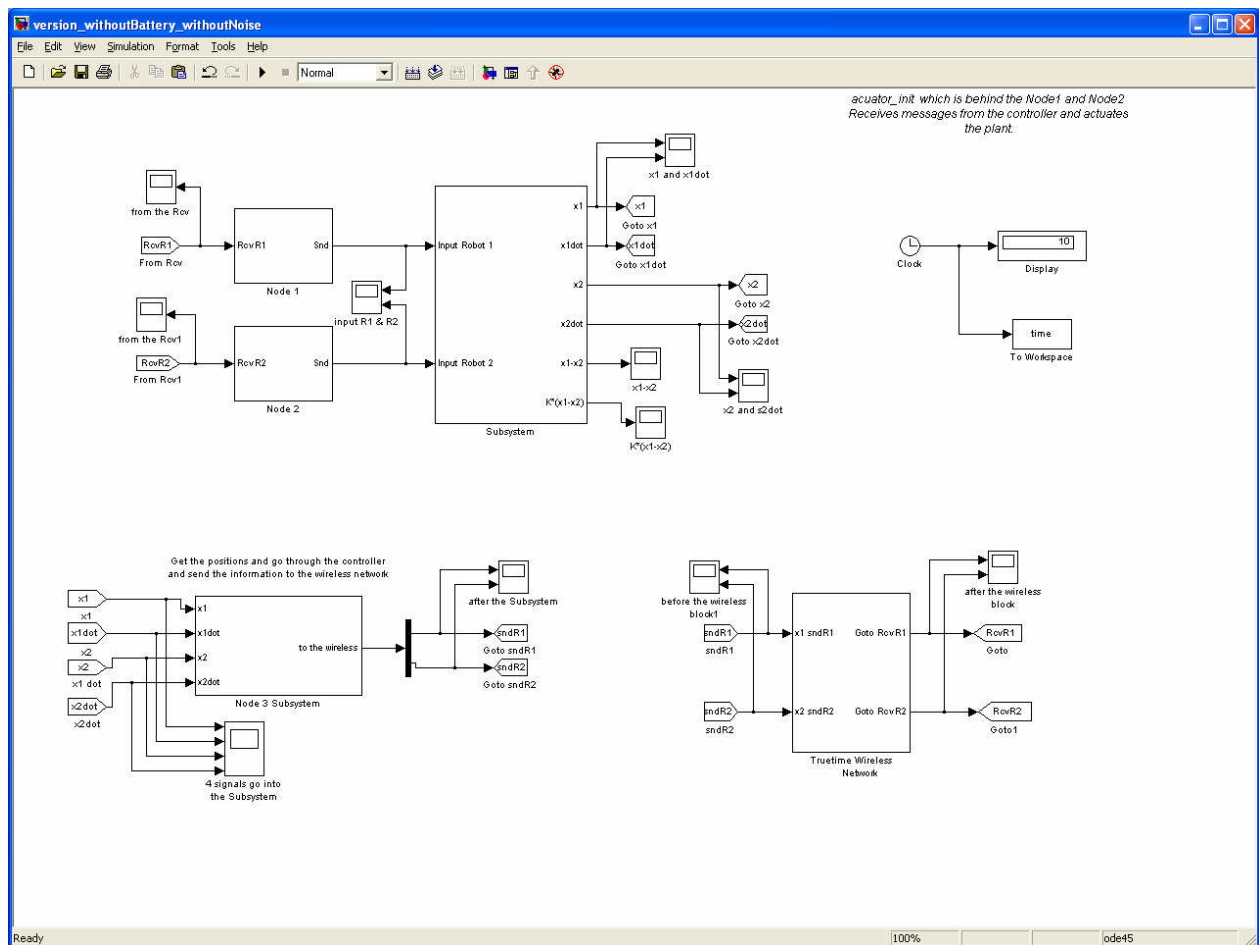


Фигура 44 Сравнение на входни и изходни сигнали

Приложението, което се използва за да се симулира безжична комуникация, е приложение съвместимо с Matlab/ Simulink – TrueTime [41][42].

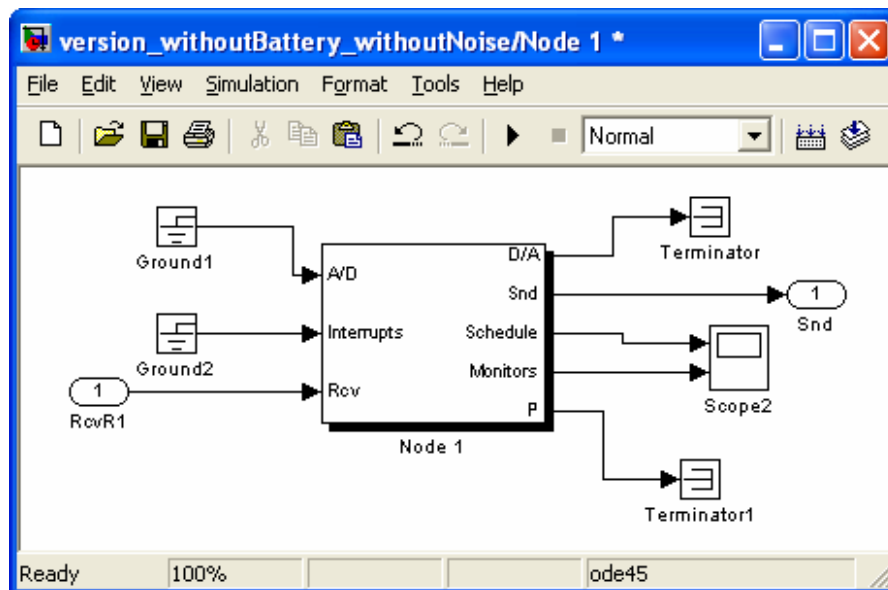
Имайки модела може да приложим симулация за безжична комуникация, използвайки приложението TrueTime, с основа Matlab/ Simulink.

На фиг 45. е показаната схема на модела за два робота, използвайки TrueTime.



Фигура 45 Симулация на безжична комуникация между два робота

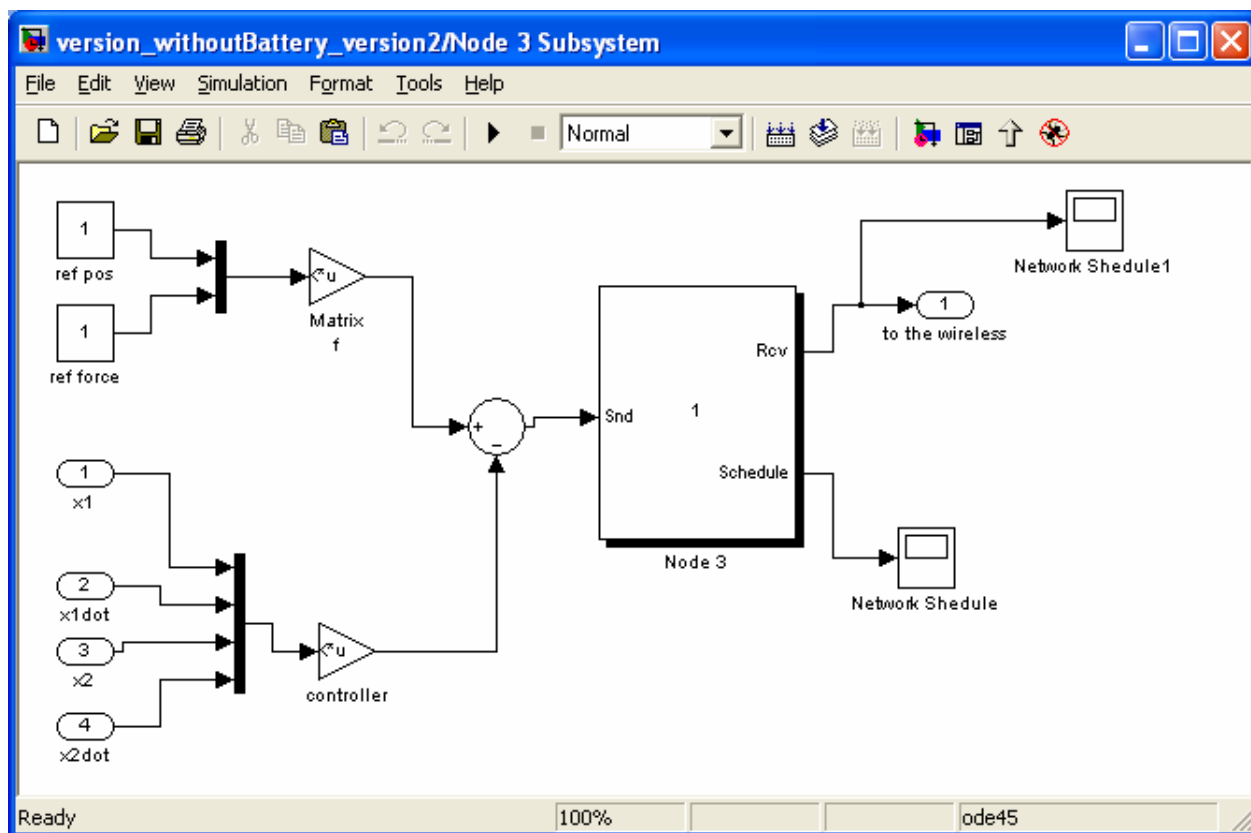
Предаването от аналогов към цифров сигнал е направено по следната схема – фиг. 46:



Фигура 46 Аналогово цифрово преобразуване

Имайки схемата за предаване на данни с безжична комуникация можем да свържем модела с тази схема. Използват се четири възела по два за всеки от роботите. Тези четири възела се свързват към локалната система на модела, където се намира симулатора. Отделно има един допълнителен възел, чрез който да се прехвърлят данните и до командния компютър.

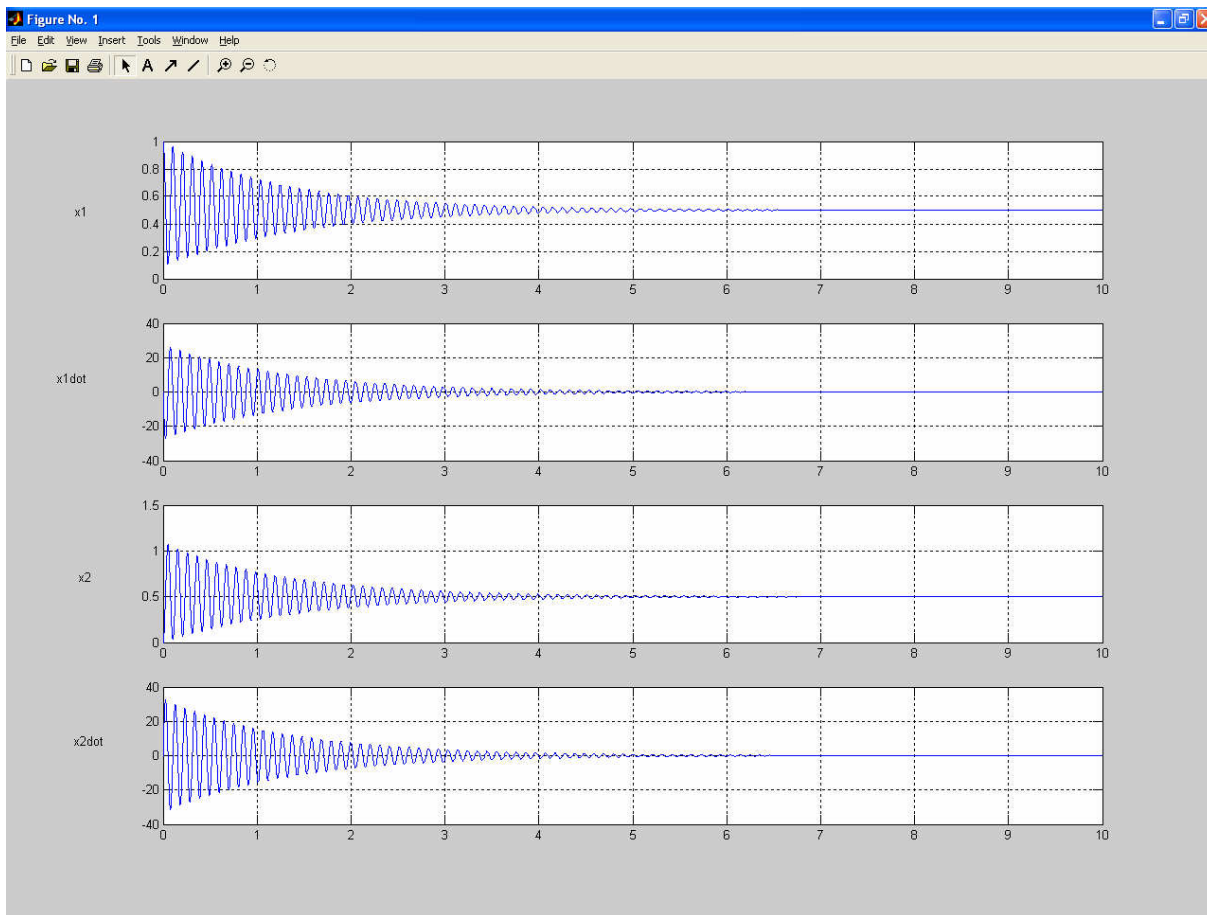
Това което е зад локалната система 3 е показано на фиг. 47 :



Фигура 47 Безжична връзка

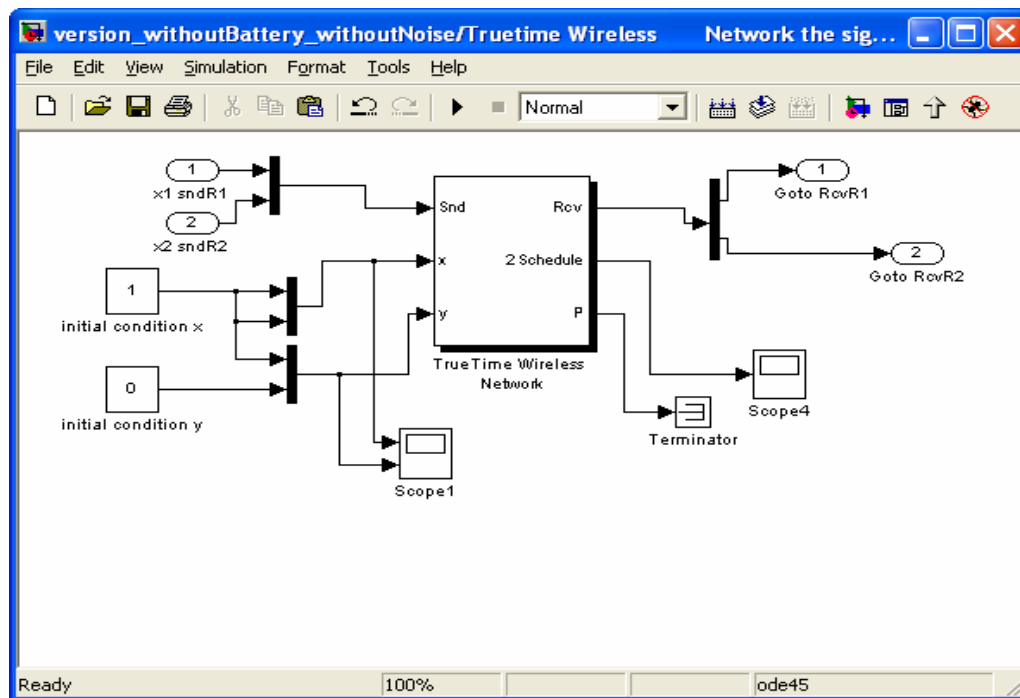
Сигналите, които влизат и се предават в тази локална система са показани на фиг. 48. От фигурата се вижда, че сигналът не отива веднага към устойчиво състояние. От това може да се заключи, че закъснението при предаването на данни, което се очаква да бъде достатъчно голямо, така че вторият робот да не получи навреме сигнал от командния компютър и да се получи закъснение при тръгването. Това закъснение може доведе до изпускане на пренасяния обект,

поради това че първият робот започва движението си по бързо от втория.



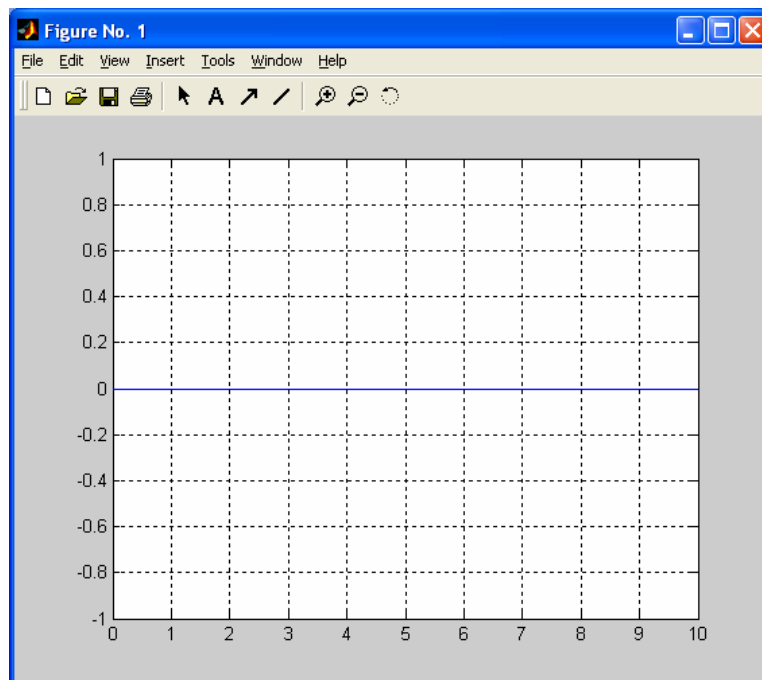
Фигура 48 Изходни сигнали

На фиг. 49. е показана и блок-схемата на блока за безжична комуникация.

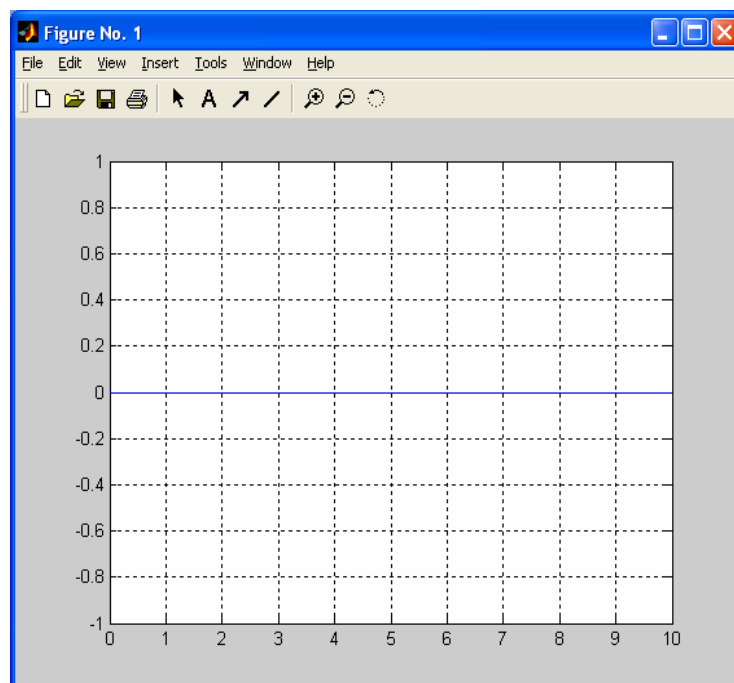


Фигура 49 Безжична комуникация

Сигналите, които се получават в симулатора от двата робота са следните, фиг. 50, фиг. 51:

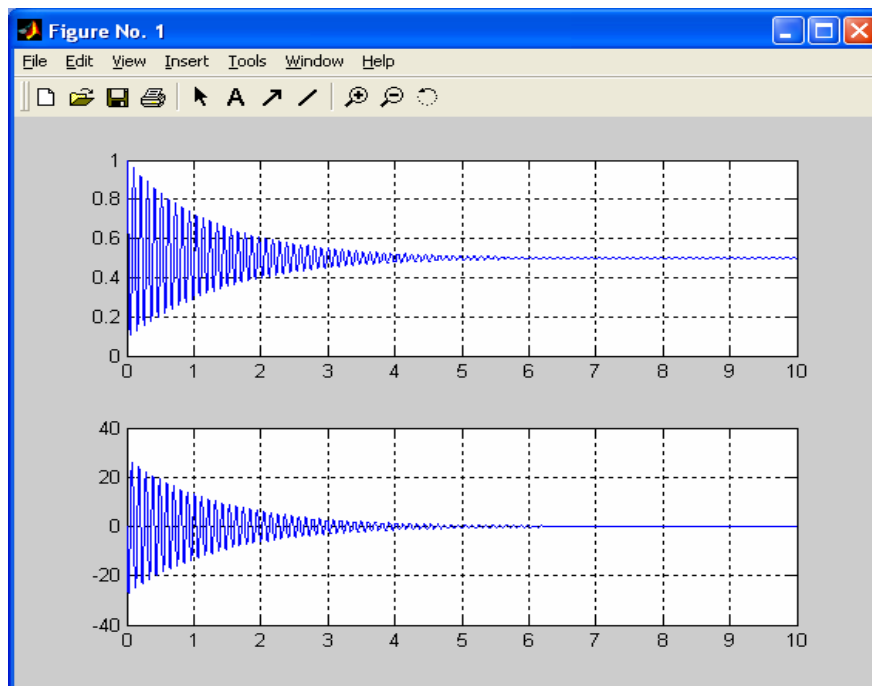


Фигура 50 Изходен сигнал 1



Фигура 51 Изходен сигнал 2

След симулатора се получават следните данни от Робот 1 – фиг. 52:



Фигура 52 Изходен сигнал

5 Глава 5 Заключение

В разработената дисертация прилагам алгоритми за групово управление на мобилни микро работи както в реално време така и офлайн. Разработени са алгоритми за обхождане на повърхност, разгледани са случай при които група от работи изпълняват поставени задачи, синхронизация на работи в реално време и комуникацията помежду им.

В разработената дисертация са разгледани алгоритми за групово управление на мобилни микро роботи. Разглеждам проблема както като симулационна среда, така и при прилагането на алгоритмите в реално време с роботи от тип e-Ruck. В Глава 3 съм описала математичен модел за реализацията на алгоритмите за обхождане на повърхност. Също в Глава 3 е разгледано различно поведение на два работа пренасящи товар. При обхождане на повърхност винаги биха могли да изникнат проблеми от счупено колело до грешка при пренасянето на данни, поради тази причина също така са разгледани и ситуации при които такива ситуации биха могли да се случат.

При микророботите трябва да се вземат предвид размерите, сложността на работната среда, автономно захранване за ефекторната система, ограничеността на датчици и двигатели, поради тази причина е направена оптимизация на разгледаните алгоритми

Описаните дотук стратегии и дадените примери очертават насоките в развитието на методите за групово управление на мобилни микророботи. С тяхното развитие се преминава от научни изследвания и игрови ситуации към ползуването на групи мобилни микророботи, управлявани синхронизирано с обща цел в реалния свят, например за изследване на околната среда, наблюдения във вредни условия или

недостъпни места, при откриване на пострадали от природни бедствия или спасяване на заложници на терористи и др.

Имайки предвид наразстващата технология разработените алгоритми могат да бъдат доразвити и адаптирани за необходимите цели.

Описаните стратегии за поведение при различни ситуации са аргументирани и са приложени примери.

6 Литература

[1] R. Beekers, O. E. Holland, and J.-L. Deneubourg, —From local actions to global tasks: Stigmergy and collective robotics, □ in Fourth Workshop on Artificial Life, B. R. and M. P., Eds. Boston, MA, USA: The MIT Press, 1994, pp. 181–189.

[2] A. Martinoli, A. J. Ijspeert, and F. Mondada, —Understanding collective aggregation mechanisms: From probabilistic modelling to experiments with real robots, □ Robotics and Autonomous Systems, vol. 29, pp. 51–63, 1999.

[3] A. Martinoli, A. J. Ijspeert, and L. M. Gambardella, —A probabilistic model for understanding and comparing collective aggregation mechanisms, □ in Fifth European Conf. on Artificial Life, ser. Lectures

Notes in Computer Science, D. Floreano, F. Mondada, and J.-D. Nicoud, Eds., Lausanne, Switzerland, 1999, pp. 575–584.

[4] O. Holland and C. Melhuish, —Stigmergy, self-organisation, and sorting in collective robotics, □ Artificial Life, vol. 5, pp. 173–202, 1999.

[5] A. K. Das, R. Fierro, V. Kumar, J. P. Ostrowoski, J. Spletzer, and C. J. Taylor, —A vision-based formation control framework, □ IEEE Trans. on Robotics and Automation (Special Issue on Advances in Multi-Robot Systems), vol. 18, no. 5, pp. 813–825, 2002.

[6] B. P. Gerkey and M. J. Matarić, —Sold!: Auction methods for multirobot coordination, IEEE Trans. on Robotics and Automation (Special Issue on Advances in Multi-Robot Systems), vol. 18, no. 5, pp. 758–768, 2002.

[7] C. R. Kube and E. Bonabeau, —Cooperative transport by ants and robots, Robotics and Autonomous Systems, vol. 30, no. 1–2, pp. 85–101, 2000.

[8] L. E. Parker, —Lifelong adaptation in heterogeneous multi-robot teams: Response to continual variation in individual robot performance, □ Autonomous Robots, vol. 8, pp. 239–267, 2000.

[9] M. J. Matarić, —Interaction and intelligent behavior, □ Dept. of Electrical Engineering and Computer Science, MIT, Cambridge, MA, □

Doctoral Dissertation, 1994.

[10] J. Fredslund and M. J. Matarić, —General algorithm for robot formations using local sensing and minimal communication, □ IEEE Trans. on Robotics and Automation (Special Issue on Advances in Multi-Robot Systems), vol. 18, no. 5, pp. 837–846, 2002.

[11] I. D. Kelly and D. A. Keating, —Flocking by the fusion of sonar and active infrared sensors on physical autonomous mobile robots, □ in Third Conf. on Mechatronics and Machine Vision in Practice, vol. 1, Gaimardes, Portugal, 1996, pp. 1–4.

[12] A. T. Hayes, A. Martinoli, R. M. Goodman, H. T. Nagle, J. W. Gardner, and K. Peresaud, —Distributed odor source localization, □ IEEE Sensors (Special Issue on Artificial Olfaction), vol. 2, no. 3, pp. 260–271, 2002.

[13] A. T. Hayes, A. Martinoli, and R. M. Goodman, —Swarm robotic odor localization: Off-line optimization and validation with real robots, □ Robotica (Special Issue on Bio-Inspired Robotics), vol. 21, pp. 427–441, 2003.

[14] A. Billard, A. J. Ijspeert, and A. Martinoli, —A multi-robot system for adaptive exploration of a fast changing environment: Probabilistic modelling and experimental study, □ Connection Science (Special Issue on Adaptive Robots), vol. 11, no. 3/4, pp. 359–379, 1999.

- [15] B. Yamauchi, —Decentralized coordination for multi-robot exploration, □ Robotics and Autonomous Systems, vol. 29, no. 1, pp. 111–118, 1999.
- [16] I. Wagner, M. Lindenbaum, and A. Bruckstein, —Distributed covering by ant-robots using evaporating traces, □ IEEE Transactions on Robotics and Automation, vol. 15, no. 5, pp. 918–933, 1999.
- [17] T. Weigel, J.-S. Gutmann, M. Dietl, A. Kleiner, and B. Nebel, —Cs freiburg: Coordinating robots for successful soccer playing, □ IEEE Trans. on Robotics and Automation (Special Issue on Advances in Multi-Robot Systems), vol. 18, no. 5, pp. 685–699, 2002.
- [18] G. Beni and J. Wang, —Swarm intelligence, □ in Seventh Annual Meeting of the Robotics Society of Japan, Tokyo, Japan, 1989, pp. 425–428.
- [19] A. Martinoli, G. Theraulaz, and J.-L. Deneubourg, —Quand les robots imitent la nature, □ La Recherche, vol. 358, pp. 56–62, 2002.
- [20] E. Bonabeau, M. Dorigo, and G. Theraulaz, Swarm Intelligence: From Natural to Artificial Systems, ser. SFI Studies in the Science of Complexity. New York, NY: Oxford University Press, 1999.
- [21] W. Agassounon, A. Martinoli, and R. M. Goodman, —A scalable, distributed algorithm for allocating workers in embedded systems, □ in IEEE Conf. on Systems, Man and Cybernetics SMC-01, Tuscon, AZ, USA, 2001,

pp. 3367–3373.

[22] K. Lerman, A. Galstyan, A. Martinoli, and A. J. Ijspeert, —A macroscopic analytical model of collaboration in distributed robotic systems, □ *Artificial Life*, vol. 7, no. 4, pp. 375–393, 2001.

[23] A. Martinoli and K. Easton, —Modeling swarm robotic systems, □ in *Eighth International Symp. on Experimental Robotics ISER-02*, ser. Springer Tracts in Advanced Robotics (2003), B. Siciliano and P. Dario, Eds., Sant’Angelo d’Ischia, Italy, 2002, pp. 297–306.

[24] —, —Optimization of swarm robotic systems via macroscopic models, □ in *Second International Workshop on Multi-Robot Systems*, A. C. Schultz, L. E. Parker, and F. E. Schneider, Eds., Washington, DC, USA, 2003, pp. 181–192.

[25] W. Agassounon, —Modeling artificial, mobile swarm systems, □ *Doctoral Dissertation*, California Institute of Technology, 2003.

[26] A. Martinoli, K. Easton, and W. Agassounon, —Modeling of swarm robotic systems: A case study in collaborative distributed manipulation, □ *International Journal of Robotics Research* (Special Issue on Experimental Robotics), vol. 23, no. 4, pp. 415–436, 2004.

[27] W. Agassounon, A. Martinoli, and K. Easton, —Macroscopic modeling of aggregation experiments using embodied agents in teams of constant and

time-varying sizes, □ Autonomous Robots (Special Issue on Swarm Robotics), vol. 17, no. 2-3, pp. 163-191, 2004.

[28] K. Easton and J. Burdick, —A coverage algorithm for multi-robot boundary inspection, □ in IEEE International Conference on Robotics and Automation ICRA-05, Barcelona, Spain, 2005.

[29] C. S. Orloff, —A fundamental problem in vehicle routing, □ Networks, vol. 4, pp. 35-64, 1974.

[30] K. Williams: Multi Robot Systems: Modeling Swarm Dynamics and Designing Inspection Planning Algorithms. 2006 California University of Technology.

[31] D. Ahr and G. Reinelt, —New heuristics and lower bounds for the min-max k-chinese postman problem, □ in Algorithms-Esa 2002, Proceedings, ser. Lecture Notes in Computer Science, 2002, vol. 2461, pp. 64-74.

[32] G. N. Frederickson, M. S. Hecht, and C. E. Kim, —Approximation algorithms for some routing problems, □ SIAM Journal on Computing, vol. 7, no. 2, pp. 178-193, 1978.

[33] GraphViz Visualization Software.

[34] Webots real time robot environment .

[35] Т. Митева, Д. Карастоянов: Групово Управление на Мобилни

Микророботи, ПРАКТРО 2005, Варна, България.

[36] Карастоянов Д. Н. и др., Мобилни роботизирани системи за работа във вредни среди., Договор с НФНИ-МОН № ТН-801/98 г., доклади на Национална конференция —Роботика и мехатроника 1999□, 23-25 юни 1999, Известия на НТСМ, бр. 5, 1999

[37] Т. Митева, Д. Карастоянов: Behavior based mobile robots and movements, October 2005, National Conference on Robotics, Dryanovo.

[38] Д. Карастоянов, Т. Митева, Б. Соколов, С. Стоенчев: Методи за групово управление на Мобилни Микро роботи, Септември 2008, 18та Национална Конференция по Роботика и Мехатроника, Варна, България.

[39] R. Buechi, State Space Control, LQR and Observer.

[40] B. Anderson, J. Moore: Optimal Control: Linear Quadratic Methods.

[41] H. Bode: Analyze and Simulation of dynamic systems

[42] LUND University: TrueTime: Simulation of networked and Embedded Control Systems.

[43] Заманов В., Карастоянов Д., Сотиров З., Механика и управление на роботите, С, Литера принот, 1993

[44] Карастоянов Д., Харачерева Г., Преглед на съществуващи крачеши мобилни роботи., Проблеми на техническата кибернетика и

роботиката, бр. 33, С, 1991, стр. 42-48

[45] Карастоянов Д. Н., Захариев Р. З., Мобилна роботизирана система с манипулационни възможности за работа във вредни и опасни среди., X Национална конференция с международно участие “Автоматизация на дискретното производство – АДП 2000”, май 2000 г., стр. 245-251

[46] Burrows M., Yeomands D., Magnetically attached vehicles for a reactor pressure vessel inspection., Proceedings of Conference on Remote Techniques for Nuclear Plant, British Nuclear Energy Society, Stanford-upon-Avon, England, May, 1993, pp 152-156

[47] Bergander A. an other., Mobile cm^3 -microrobots with tools for nanoscale imaging and micromanipulation., <http://www.microrobotics.epfl.ch/pdf>

[48] <http://www.epfl.ch>

[49] Halme A., Jakubik P., Schönberg T., Vainio M.: The concept of robot society and its utilization, Proceedings of the 1993 IEEE International Workshop on Advanced Robotics, Tsukuba, Japan, pp. 29-35, 1993

[50] Arkin, R.C. and Balch, T.R. 1998. Cooperative Multiagent Robotic Systems AI-based Mobile Robots: Case Studies of Successful Robot Systems. Kortenkamp, D., Bonasso, R.P. and Murphy, R. (eds). MIT Press.

[51] Parker L.E.: Multi-Robot Team Design for Real-World Applications,

Distributed Autonomous Robotic Systems 2, Asama H., Fukuda T., Arai T., Endo I., Asama, (Eds.), Springer-Verlag, Tokyo, pp. 91 - 103, 1996.

[52] Parker, L.E. 1999. Adaptive Heterogeneous Multi-Robot Teams, *Neurocomputing, special issue of NEURAP '98: Neural Networks and Their Applications 1999*, vol. 28, pp. 75-92.

[53] Миланов С. Мрежи: Wireless LAN (безжични комуникационни устройства),

http://www.comexgroup.com/communications/lan_all/wlan.htm, 02/24/2004

[54] Миланов С. Интерфейс Bluetooth, www.comexgroup.com/communications/bluetooth.htm, 02/24/2004

[55] Atmel Corporation, The Bluetooth Wireless Technology, White paper, http://www.atmel.com/dyn/products/product_card.asp?part_id=2205, 2000

[56] Atmel Corporation, Bluetooth General Information, White paper, http://www.atmel.com/dyn/products/product_card.asp?part_id=2205, 2000

[57] Kinney P., ZigBee Technology: Wireless Control that Simply Works, Home Toys Article, <http://hometoys.com/htinews/oct03/articles/kinney/zigbee> .htm, October 2003

[58] ZigBee Alliance, ZigBee Overview, <http://www.zigbee.org/documents/ZigBeeOverview4.pdf>, September 2003

- [59] ZigBee Alliance ,The ZigBee Buzz Is Growing: New Low-Power Wireless Standard Opens Powerful Possibilities, http://www.zigbee.org/documents/ZigBee_CES_January_2004_FINAL.pdf, January 2004
- [60] ZigBee Alliance, FAQ's, <http://www.zigbee.org/about/faqs/index.asp>, October, 2003

7 Таблица на използвани фигури

фигура 1 Летящ Микроробот mFR.....	20
фигура 2 Летящо насекомо Robofly	21
фигура 3 Работа в клетка.....	22
Фигура 4.....	34
Фигура 5 Примерна среда за обхождане на повърхност, възникващи проблеми	39
Фигура 6 Работна среда.....	41
Фигура 7 Области на видими зони за обхождане.....	42
Фигура 8 Външни сегментни граници.....	43
Фигура 9 Райони за инспекция.....	46
Фигура 10 Локални компоненти	47

Фигура 11 Конструкция на задължителни криви и точки за обхождане	47
Фигура 12 Презентация на напълно свързан не директен граф.....	50
Фигура 13 Прилагане на GraphViz приложение за конструиране на граф във работна среда.....	51
Фигура 14 Приложен Дейстра алгоритъм за намиране на най-краткото разстояние от първа до последна точка на графа.....	54
Фигура 15 Прилагане на Дейстра алгоритъм за намиране на най-краткото разстояние от зададена начална точка.....	55
Фигура 16 Три пъти приложен алгоритъм за намиране на най краткото разстояние.....	57
Фигура 17 Разпределение на път.....	58
Фигура 18 Първи Webots робот.....	60
Фигура 19 Страни и координати на работната среда.....	63
Фигура 20 Работна среда представена във Webots	65
Фигура 21 Координатна система и примерен обикновен робот.....	66
Фигура 22 Конфигурация на колелата на примерен обикновен робот .	67
Фигура 23 Примерен обикновен робот с тяло и колела с форма на цилиндър	69
Фигура 24 e-Ruck робот в работна среда.....	70

Фигура 25 Прикачена камера на e-Ruck робот	71
Фигура 26 Работна среда.....	72
Фигура 27 Работна среда с вграден граф	73
Фигура 28 Разграфяване на граф при наличие на 4 робота.....	77
Фигура 29 Приложен алгоритъм на Дейстра.....	78
Фигура 30 Среда с три робота, пълно обхождане	81
Фигура 31 Среда за обхождане при възникнал проблем при робот със синя линия на обхождане.....	81
Фигура 32 Ситуация при присвояване на път от страна на друг робот	82
фигура 33 Връзки между Bluetooth устройства в piconet с едно устройство slave (a), няколко устройства slave (b) и свързване от типа scatternet	92
Фигура 34 Подходи.....	105
Фигура 35 Опростена блокова схема.....	108
Фигура 36 Изходен сигнал.....	109
Фигура 37 Изходен сигнал при различни стойности.....	111
Фигура 38 Симулация с два робота при включен контролер	116
Фигура 39 Изходни сигнали	117
Фигура 40 Симулация при наличие на бял гаусов шум	119
Фигура 41 Изходен сигнал.....	120

Фигура 42 Изходен сигнал на скорост.....	121
Фигура 43 Сравнение на входни и изходни сигнали	121
Фигура 44 Симулация на безжична комуникация между два робота.	123
Фигура 45 аналогово цифрово преобразуване.....	124
Фигура 46 Безжична връзка	125
Фигура 47 Изходни сигнали	126
Фигура 48 Безжична комуникация.....	127
Фигура 49 Изходен сигнал.....	128
Фигура 50 Изходен сигнал.....	128
Фигура 51 Изходен сигнал.....	129