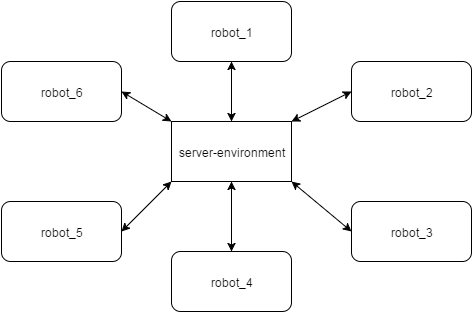
# Математико-компьютерная модель группировки спасательных роботов

Программа предназначена для моделирования поведения и группового взаимодействия спасательной группировки. «agent-robot» моделирует отдельного робота, «server-environment» - моделирует среду и каналы связи, библиотека utils описывает протокол взаимодействия между роботами, состояния роботов, команды.

Программное обеспечение является распределенным: каждый отдельный робот запускается на отдельной JAVA машине и является самостоятельной программой.



Роботы обмениваются сообщениями через сервер, что позволяет моделировать качество канала и моделировать помехи при обмене сообщениями между роботами.

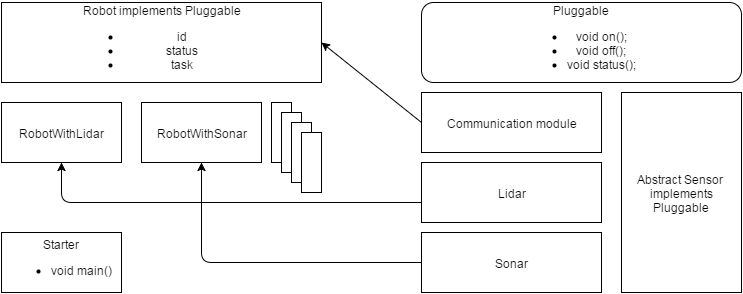
Пульт оператора реализован как Web приложение, и может быть запущен на любом устройстве с предустановленным браузером.

С пульта оператора можно получить информацию о любом роботе и группировке в целом.



## Структура модуля «agent-robot»

Робот представляет собой самостоятельную программу, запущенную на отдельной Java машине. Упрощенно схема программы робота представлена на рисунке.



Каждый робот реализует интерфейс Pluggable – содержащий методы on(), off(), status(). Эти методы являются базовыми для роботов и его внутренних устройств таких как сенсоры.

Каждый робот включает следующий обязательный набор полей:

* Int id;
* AgentStatus status;
* AgentTasks task.

Идентификатор робот однозначно идентифицирует его в системе. Статус робота позволяет определить его состояние: включен, выключен, потеря работоспособности, частично неисправен. Статусы состояния робота описаны в файле AgentStatus.java. AgentTask представляет собой задачу выполняемую роботом. Также у робота есть поле communicationModule – представляющее собой экземпляр класса модели средства коммуникации робота, через который он производит общение с остальной системой. Для реализации взаимодействия между роботами (подключения и отправки сообщений) была выбрана технология WebSocket.

От базового класса робота наследуются его более частные реализации, например, робот, оснащенный лазерным сканирующим дальномером. От базового класса робота он отличается наличием специального сенсора и специфическим для него поведением.

Все сенсоры робота наследуются от базового класса AbstractSensor реализующего интерфейс Pluggable. Соответственно любой из сенсоров робота может быт включен, выключен, неисправен и т.п., что позволяет реализовывать различные сценарии поведения робота при изменении его конфигурации.

Входными данными для работы программы является:

данные полученные от программных моделей роботов: строки в формате JSON оформленные в соответствии с протоколом сообщений,

данные полученные от пульта управления оператора: строки в формате JSON оформленные в соответствии с протоколом сообщений.

## Структура модуля «server-environment»

Основная задача серверной части модели – это моделирование среды окружающей робота, моделирование канала связи между роботами и передача сообщений между ними.

Так как процесс обмена сообщениями между роботами и сервером реализован на базе технологии WebSocket, все соединения между роботами и сервером осуществляются через Socket.

Сообщения от робота к серверу могут быть двух типов:

* информация о действиях робота в среде и общение сенсоров со средой;
* сообщения, которыми обмениваются между собой роботы.

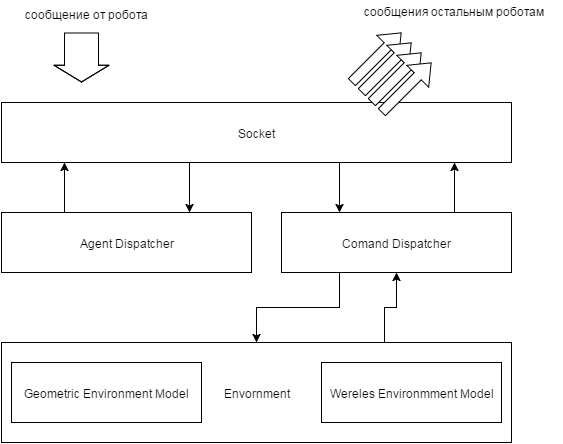


Рисунок - схема работы приложения server-environment.

Принципиальную важность играют сами сообщения, которыми обменивается робот и их формат.

В общем виде сообщение между роботами имеют следующий формат:

{

id:[integer],

command: [command] ,

data: {}

}

Данные передаются в формате JSON представляющий собой текстовый формат обмена данными, сообщение представляет собой набор полей и значений. Каждое сообщение должно включать в себя id робота который отправляет сообщение, команду, из списка команд группировки и специальное поле data куда включается вся вспомогательная информация.

Сообщения, которыми робот обменивается непосредственно со средой так же проходят через сокет, и используют то же соединение что и коммуникационный модуль но непосредственно сам модуль коммуникации при этом не затронут. Обмен этими сообщениями представляет собой уровень абстракции над работой сенсоров робота. Каждое измерение, сделанное сенсором робота или движение внутри среды, представляется как сообщение от робота, серверу среды.

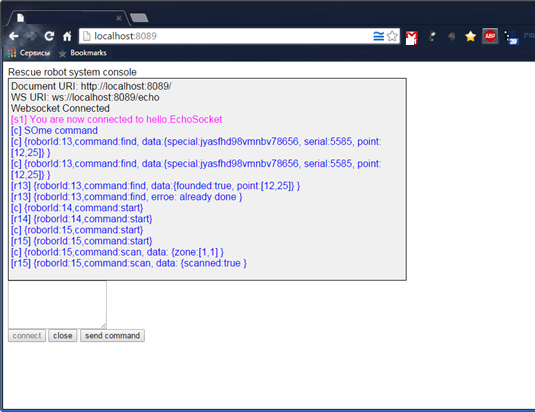
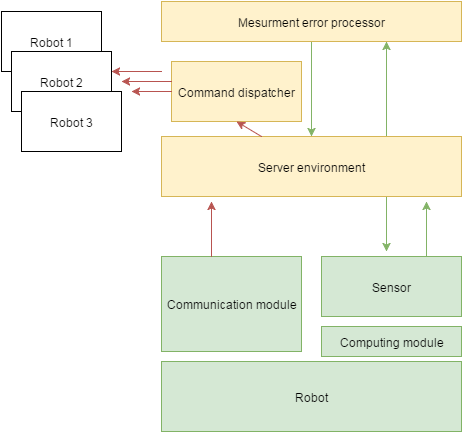


Рисунок - скриншот работы консоли управления

Среда возвращает роботу ответ на его действия и изменяет свое состояние, в ответ на изменение среды робот производит адекватные этому изменению действия и также изменяет свое состояние. Такой подход позволяет моделировать погрешность сенсоров робота и возвращать роботу искаженные данные о среде.

Получив сообщение от робота о каком-либо измерении сервер производит вычисление результата измерения и возвращает его сенсору робота. Сенсор получив данные передает их в вычислительный блок робота где производится обработка и интерпретация полученных данных. На рисунке блоки робота обозначены зеленым цветом, блоки сервера среды желтым, остальные роботы белым. Сообщения, отправляемые другим роботам обозначены красными стрелками, а сообщения между сенсорами робота и средой зелеными.



## Формирование знаний группировки о мире

Так как разные роботы обладают разными вычислительными мощностями и разными наборами сенсоров, то они имеют и разные возможности по накоплению, обработке и наполнению картины мира роботов информацией.

Совокупное знание роботов о мире формирует знание всей группировки. Причем доступно оно будет с той точностью и разрешающей способностью, которой обладает самый высоко точный робот. Однако знание роботов о среде будет неоднородно вследствие неоднородности, собираемой ими информации.

Совмещение информации должно происходить таким образом, что при передаче данных между роботами перед совмещением данных в свою собственную картину мира, робот должен производить конвертацию и фильтрацию входящих данных в соответствии с форматом своей внутренней модели.

Для упрощения обмена информации о среде (в данном случае карты), территория на которой проводится спасательная операция разбивается на зоны фиксированного размер. Зоны кодируются двумя числами являющиеся порядковыми номерами зон по двум осям.

В процессе работы роботы обновляют участки карты, на которой они работают. Каждый участок принадлежит определенной зоне, время последнего обновления зоны сохраняется роботом. Через фиксированные промежутки времени роботы обмениваются сообщениями со списком зон и датой их обновления. Если робот обнаруживает в чужом списке более актуальную версию зоны он запрашивает ее у другого робота. В случае если робот обнаруживает в чужом списке более старую версию зоны от отправляет этому роботу сообщение с обновленным участком зоны. Такой подход позволяет уменьшить количество передаваемой по сети информации передавая актуальные данные о среде адресно каждому роботу.

Передаваемая роботами информациями обладает следующими характеристиками:

* полнота;
* актуальность;
* подробность или разрешение.

Полнота определяет на сколько полной информацией о зоне располагает робот. В процессе своей работы робот может обладать информацией только о части зоны. Актуальность характеризуется датой последнего обновления информации о зоне. Разрешение определяется насколько точные сенсоры обновляющего участок карты.

Передавая информацию о среде робот, обладающий более точными сенсорами и картой в высоком разрешении роботу, использующему карты с низким разрешением, необходимо произвести фильтрацию данных (преобразование карты). Преобразование карты производится объединением числа ячеек кратного разрешению второго робота.

## Распределение задач

В процессе выполнения спасательной операции роботам необходимо выполнять слаженные скоординированные действия. Для этого роботы формируют общий список задач, упорядоченных по приоритету.

При появлении новой задачи ей назначается приоритет, соответствующий этому типу задач. Робот помещает ее в общий список задач.

Задача имеет следующий формат:

* заголовок задачи состоящий из
  + id задачи
  + типа задачи
  + приоритета задачи
* описание задачи, включающее в себя
  + объект задачи
  + описание условий выполнения задач.

Для типовых задач объект или условия выполнения могут отсутствовать.

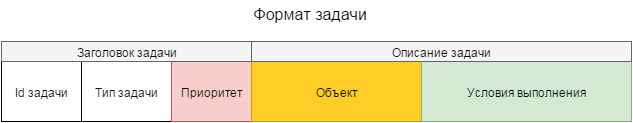


Рисунок –формат задачи.

Список сортируется в соответствии с приоритетом задач. В зависимости от типа задачи ее приоритет может расти или уменьшаться со временем. Роботы начинают аукцион между всеми роботами группировки на выполнение самой верхней задачи из списка. Каждый робот оценивает стоимость выполнения этой задачи для себя и объявляет цену. Робот, объявивший самую низкую цену, забирает задачу. Задача удаляется из списка и аукцион продолжается до тех пор, пока не будут разобраны все задачи.

Если робот не может выполнить задачу потому что она не подходит ему по типу выполняемых задач или робот не имеет прав на взаимодействие с данным типом объектов или условия выполнения для него недостижимы, он игнорирует эту задачу. Спустя фиксированный отрезок времени задача помечается как не выполненная ее приоритет пересчитывается в соответствии с правилами изменения приоритета для данного типа задач, а аукцион продолжается дальше. По окончанию круга аукциона, задача возвращается в список задач, список сортируется, и аукцион повторяется.

При достижении определенных условий, например, падения приоритета ниже порога n, задача удаляется из списка.

Для определенного типа задач, при выставление их на аукцион возможно изменение списка выполнения операций роботов и прерывания некоторых задач. Это важно для переключения роботов на выполнение чрезвычайных и неотложных задач, например, по предотвращению развития катастрофы.

## Построение карты

Определение пересечения луча с границей объекта в модели реализовано через нахождение пересечения отрезков. Отрезки в модели могут быть представлены с помощью параметрических уравнений:

Где x, y – координаты искомой точки, x1,y1 и x2, y2 – координаты начала и конца отрезка, представляющего собой луч сканера, соответственно, а t1 параметр, обозначающий на какой части отрезка лежит точка. Аналогичным будет и уравнение искомой точки для отрезка обозначающего собой грань объекта:

Здесь x3,y3 и x4,y4 – координаты грани объекта, а t2 параметр определяющий положение точки.

Исходя из этих формул, можно составить систему уравнений:

Решая ее относительно параметра t2 получим:

Подставив в уравнение значения соответствующих координат, можно получить значение параметра t2 . Параметр t1 можно найти по формуле

Подставляя одно из полученных значений в уравнение точки пересечения можно получить ее координаты. Для того чтобы точка пересечения точно находилась на отрезках должно выполняться условие:

Выполнение этих условий гарантирует, пересечение двух отрезков представляющих собой грань объекта среды и луч лазерного сканирующего дальномера.

Для того чтобы интерпретировать координаты точек, полученных в результате сканирования, производится деление координаты точки на размер ячейки карты. В результате деления координат x и y получаются соответствующие номера ячеек карты i и j.

В зависимости от размера ячеек карты карта будет более или менее точной. На рисунке представлен пример карт построенных с помощью этого алгоритма для ячеек размером 5 единиц и 50 единиц соответственно, размер пространства в обоих случаях 600 единиц, визуализация модели пространства изображена слева.

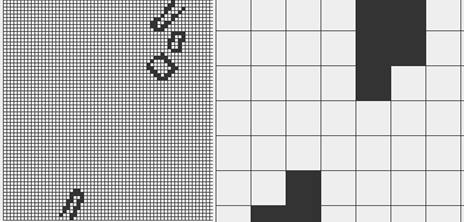


Рисунок – карты ячеек для роботов с разным разрешением сенсоров.

Для совмещения карт ячеек необходимо складывать показатели вероятностей. В процессе объединения карт также может быть проведена оценка показаний роботов.

Показания каждого робота, могут быть взвешены по какому-либо критерию, например по точности используемых сенсоров. А при сложении карт должен быть введен некоторый коэффициент, по которому будет формироваться финальное значение ячейки карты.

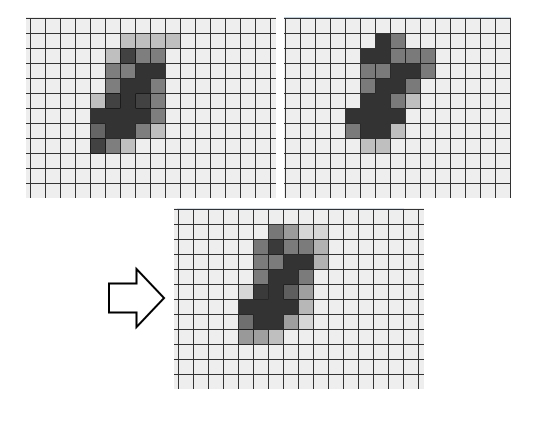


Рисунок совмещение карт роботов.

Сложение двух карт ячеек приведено на рисунке. Сверху изображены две карты ячеек, оценка вероятности возможности проезда робота через ячейку обозначена с помощью различных градаций серого – чем темнее ячейка, тем больше вероятность того, что ячейка занята препятствием. В результате совмещения двух карт получается карта с усреднёнными значениями вероятности. Часть ячеек, которые были обозначены на одной из карт как свободные для проезда стали более серыми, так как на другой карте они были обозначены как ячейки с вероятностной оценкой занятости большей, чем ноль.

На рисунке представлен скриншот визуализации работы модели во время совместного патрулирования роботами заданной зоны. Светло зелеными фигурами обозначены роботы-рабочие, красными фигурами – роботы разведчики, голубыми фигурами –препятствия. Группы препятствий могут образовывать сложную геометрию. На белой области слева представлена карта составленная роботами, представляющая собой совокупное знание роботов о мире.

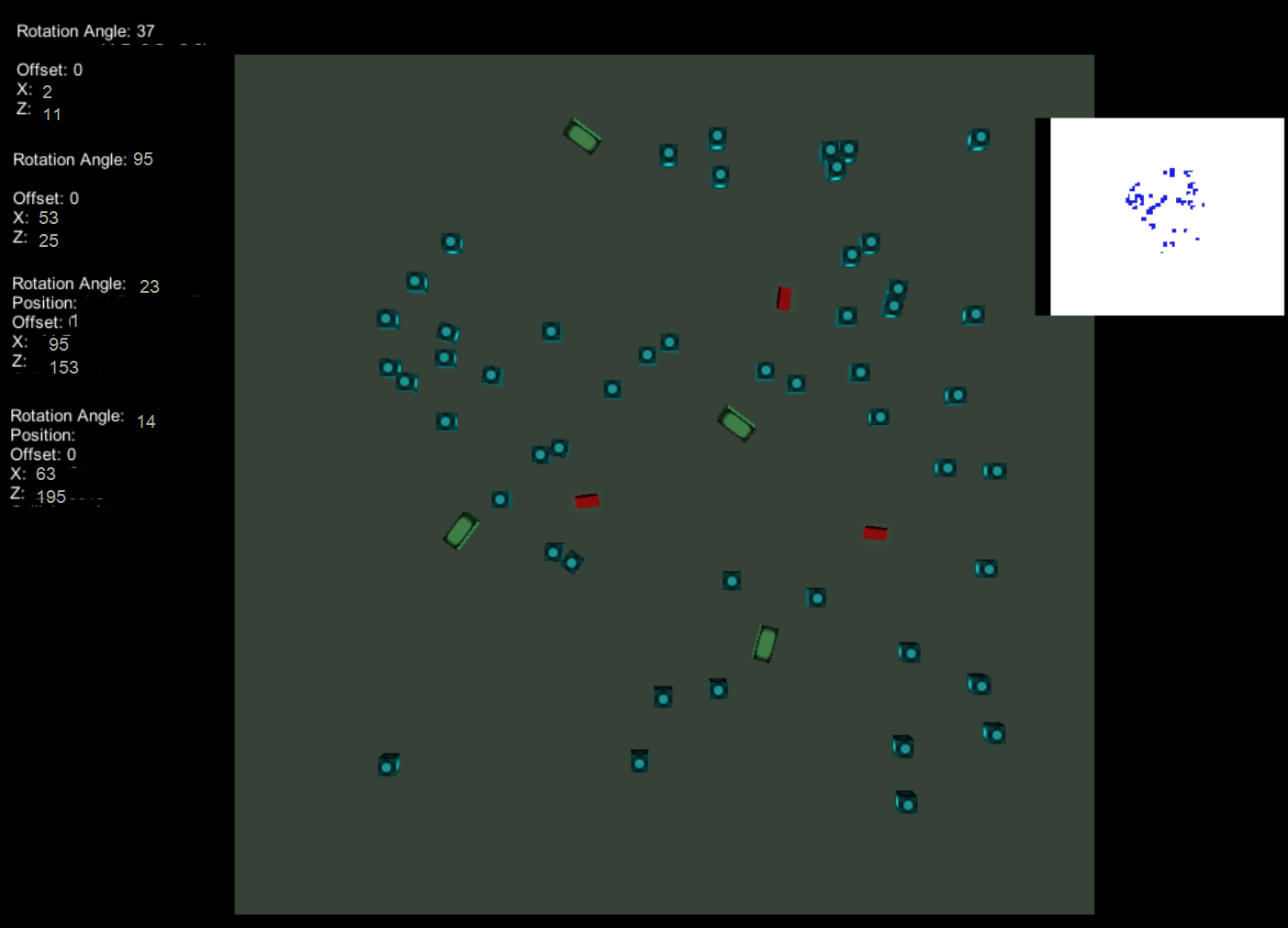


Рисунок скриншот визуализации работы модели