

УДК 681.5+004.8

**Карпова И.П.**

к.т.н, доцент, karпова\_ip@mail.ru

Национальный исследовательский университет «Высшая школа экономики»,  
Московский физико-технический институт

## **К ВОПРОСУ О ПРЕДСТАВЛЕНИИ МАРШРУТА МОБИЛЬНОГО РОБОТА НА ОСНОВЕ ВИЗУАЛЬНЫХ ОРИЕНТИРОВ\***

### **Аннотация**

В статье предлагается метод использования визуальных ориентиров для запоминания пройденного пути мобильным роботом. Описаны алгоритмы формирования описания маршрута и правила его интерпретации, позволяющие роботу вернуться в точку отправления и повторить данный маршрут. Приведены результаты экспериментов по имитационному моделированию, показывающие возможность применения разработанного метода для решения задачи фуражирования.

**Ключевые слова:** групповая робототехника, модели социального поведения, когнитивный агент, пространственная логика, задача фуражирования, грануляция информации

### **1. ВВЕДЕНИЕ**

Одним из активно развивающихся в настоящее время подходов к вопросам управления в групповой робототехнике является подход, основанный на использовании моделей социального поведения в группах роботов [1]. В рамках этого подхода исследуются различные модели социального поведения, например, когезия как стремление особей держаться вместе, контагиозное (подражательное) поведение, дифференциация функций, образование коалиций [2]. При этом за «образец для подражания» исследователи часто берут муравьев как представителей эусоциальных (истинно социальных) насекомых.

Организация социального поведения, и у муравьев в том числе, основана на внутривидовой коммуникации. Среди множества типов поведения муравьев можно выделить один, в котором роль коммуникации особенно важна, – это

---

\* Работа выполнена при финансовой поддержке РФФ, грант 16-11-00018.

фуражирование. В муравейнике есть разведчики, которые исследуют окружающее муравейник пространство в целях поиска пищи. Найдя пищу, они возвращаются в муравейник и оповещают других муравьев о месте, где была найдена пища. После этого муравьи-фуражиры отправляются к этому месту и приносят пищу в муравейник.

Данная задача разбивается на три этапа: найти «еду» (или другой искомый объект), вернуться «домой» (в точку отправления) и повторить этот путь для переноса пищи в дом. Существует большое количество исследований, посвященных решению этой проблемы, в основном, разработанных на уровне формальных моделей для групп искусственных агентов.

Для решения задачи фуражирования может использоваться механизм SLAM (Simultaneous Localization and Mapping), который предполагает построение карты местности и одновременное определение местоположения агента на этой карте и применяется для решения широкого класса задач [3]. В более специализированных методах речь идет о запоминании пути, использовании его для возвращения «домой» (homing) и повторного прохождения этого пути. Но в работах, посвященных последней группе методов, внимание, в основном, уделяется либо распознаванию образов [4], либо построению сцен [5]. Для применения любого из перечисленных методов роботы должны иметь развитую систему технического зрения, большую вычислительную мощность и каналы связи с высокой пропускной способностью для передачи построенной карты или визуального ряда, который образует маршрут, другим роботам.

Среди других подходов наиболее распространенным является использование принципа феромонов, взятого из жизни некоторых видов муравьев. Он основан на том, что муравьи при движении помечают свой путь определенными химическими веществами – феромонами, которые имеют свойство «испаряться» со временем. Муравей может уловить этот феромон и определить его концентрацию. В предлагаемых моделях агенты также условно помечают свой путь «феромонами». В дальнейшем другие агенты определяют

маршрут в соответствии с максимальной концентрацией феромонов (например, [6]). При этом для возврата в точку отправления агент может использовать маяк, расположенный около «дома» [7], GPS-навигацию для открытых пространств [8], механизм направленной коммуникации [9] или движение по собственному феромоновому следу [10].

Подход, основанный на феромонах, имеет как минимум один существенный недостаток: технически сложно реализовать феромоновый след. Другие описанные способы организации возвращения «домой» также имеют серьезные ограничения или требуют наличия дополнительных механизмов: или предполагают наличие вспомогательных технических устройств (маяков); или ориентированы на большое количество роботов (для обеспечения полного покрытия пути от «дома» до искомого объекта при использовании направленной локальной коммуникации); или требуют открытого пространства для возможности применения GPS-навигации.

С другой стороны, не все муравьи метят свой путь феромонами. Например, в ходе наблюдений за муравьями рода *Формика* (*Formicidae*) было выявлено, что они могут ориентироваться по отчетливому источнику света (солнцу или луне), по направлению колебаний света (поляризация неба в естественных условиях) и по наземным ориентирам [11]. В процессе поиска пищи муравей запоминает свое положение относительно солнца, визуальные ориентиры, мимо которых он проходит, и примерное расстояние до них. Найдя пищу и вернувшись в муравейник, он должен организовать муравьев-фуражиров для доставки пищи в муравейник. Здесь возможны варианты [12]: либо разведчик ведет фуражиров за собой, либо он объясняет им, как они могут дойти до пищи, и далее они идут туда самостоятельно. В ходе многочисленных опытов было установлено, что муравьи могут передавать сведения о маршруте другим особям в процессе тактильного или визуального контакта (например, ударяя лапками по спине другого муравья [11]).

Итак, для успешного решения задачи фуражирования робот должен уметь строить модель внешней среды на основе естественных пространственных

отношений. Для этого обычно используется псевдофизическая логика (ПФЛ): пространственная логика с нечеткими бинарными пространственными отношениями трех типов – отношения для расстояний, для направлений и отношения взаимного расположения объектов.

Пространственные логики можно условно разделить на статические и динамические. Примером применения статической логики для построения модели внешнего мира является работа Калущкой А.П. и Тарасова В.Б. [13]. Там предложен вариант построения ПФЛ с помощью обобщенных ограничений, который позволяет выводить дополнительные нечеткие пространственные ограничения, не описанные человеком явно. В качестве примера построения динамической модели можно привести работу Карпова В.Э. [14], в которой автор связывает начало системы координат с роботом и вводит набор правил определения отношений между объектами окружающей среды и роботом. Эта правила позволяют на основе логического вывода прогнозировать изменение положения объектов при выполнении роботом элементарных действий (поворотов направо / налево и шагов вперед / назад), а также их комбинаций. Однако, использование подобных механизмов в явном виде для решения задачи фуражирования затруднительно, т.к. в первом случае рассматривается статический вариант, а во второй работе решается задача прогнозирования изменения взаимного положения объектов. Кроме того, применение ПФЛ относится к задачам с высокой вычислительной сложностью.

Таким образом, существующие работы в этой области ориентированы на технически сложные устройства с большим объемом памяти и большой вычислительной мощностью. В то же время муравей использует для решения задачи фуражирования достаточно ограниченные ресурсы: «мозг» (надглоточный ганглий), содержащий примерно 500 тысяч нейронов, неточные органы чувств и канал связи с низкой пропускной способностью. Исходя из этого цель данной работы – предложить такой механизм построения динамической модели ориентации робота в окружающем пространстве, которая позволяла бы управлять перемещением робота и не требовала бы для

применения больших вычислительных мощностей и развитой сенсорики. Модель должна включать правила формирования сведений о маршруте, по которому он перемещается, и правила интерпретации этих сведений, применяя которые этот же (или другой) робот может повторить данный маршрут.

## 2. ОРИЕНТАЦИЯ РОБОТА В ПРОСТРАНСТВЕ

Определим, каким образом робот мог бы ориентироваться в пространстве. Для этого кратко перечислим те способы, с помощью которых ориентируются муравьи. У разных видов муравьев есть разные способы ориентации, но общими для большинства видов является светоконпасная ориентация и использование визуальных ориентиров [11].

На коротких маршрутах вблизи муравейника муравьи используют в основном зрительные ориентиры. На более протяженных маршрутах они включают свою врожденную систему интегрирования пути (path integration) [15]. Работа этой системы основана на одометрии и ориентации по солнцу или другому источнику света. Например, пустынные муравьи рода *Cataglyphis* считают количество пройденных шагов и запоминают свое положение относительно солнца [16]. Однако применение системы интегрирования пути приводит к накоплению ошибки, поэтому муравьи используют этот механизм при отсутствии визуальных ориентиров.

Если же говорить о возможности применения одометрии для технических устройств (учет показаний энкодеров, считающих углы поворота колес), то здесь ошибка счисления будет недопустимо большой. Качественный энкодер имеет погрешность в 0.1%, но даже при длине пути в 100 шагов (100 оборотов колес) погрешность достигнет 10%. Использовать источник света как ориентир также не представляется возможным, поэтому возьмем за основу построения пространственной модели внешней среды подход, основанный на навигации по зрительным ориентирам.

Итак, пусть на полигоне расположены неподвижные объекты – ориентиры. Робот должен уметь отличать один ориентир от другого, и для простоты

предположим, что он умеет различать цвета. Таким образом, цвет будет играть роль идентификатора объекта. Помимо этого робот умеет двигаться вперед и назад, поворачивать направо и налево, останавливаться; он видит окружающие объекты (внутри области видимости в  $180^\circ$ ) и умеет определять расстояние до них (приблизительно) и направление (относительно себя) (рис. 1). В связи с этим естественно напрашивается подход, основанный на нечетком управлении.

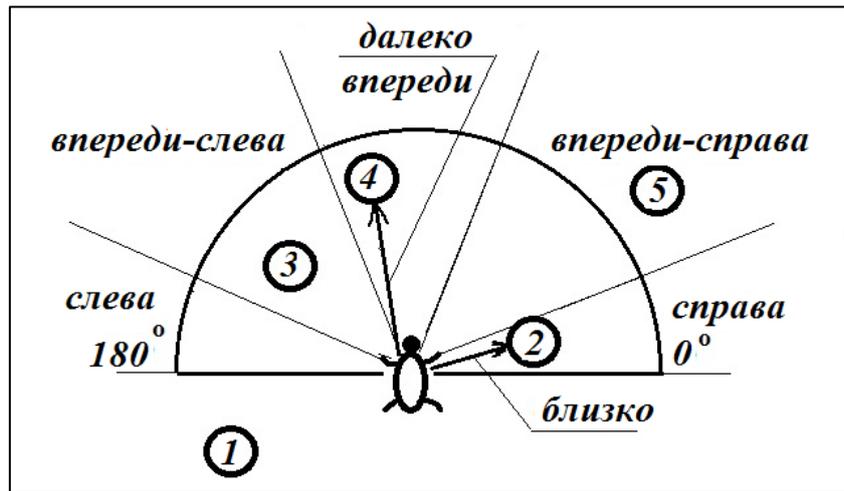


Рис. 1. Область видимости робота, определение направления и расстояния

Для решения поставленной задачи необходимо предложить формат описания пути робота-разведчика и разработать набор правил, который по этому описанию сформирует последовательность действий, позволяющую повторить путь разведчика.

### 3. ФОРМИРОВАНИЕ ОПИСАНИЯ ПУТИ И ЕГО ИНТЕРПРЕТАЦИЯ

Рассмотрим проблему формирования описания пути (запоминания пройденного маршрута). Муравей-разведчик запоминает путь приблизительно, и муравей-фуражир повторяет этот путь в общих чертах [11]. Поэтому при формировании описания маршрута имеет смысл запоминать не каждый шаг робота, а только моменты изменения обстановки: появление новых ориентиров, изменение направления на ориентиры и т.д. Таким образом, в модель вводится понятие времени, привязанное не столько к тактам (шагам робота), сколько к изменению его состояния относительно окружающей среды.

Умение различать цвета и примерно определять направления и расстояние до ориентиров позволяет роботу описывать маршрут в терминах

соответствующих лингвистических переменных. Например, маршрут, представленный на рис. 2, можно описать так: «Сначала я увидел ориентир 2 (зеленый) впереди-справа далеко; потом ориентир 2 (зеленый) был от меня справа не далеко/не близко и появился ориентир 3 (желтый) впереди-слева далеко; потом ориентир 2 (зеленый) – справа не далеко/не близко, ориентир 3 пропал из поля зрения, но появился ориентир 5 (красный) – впереди-слева далеко ...» и т.д.).

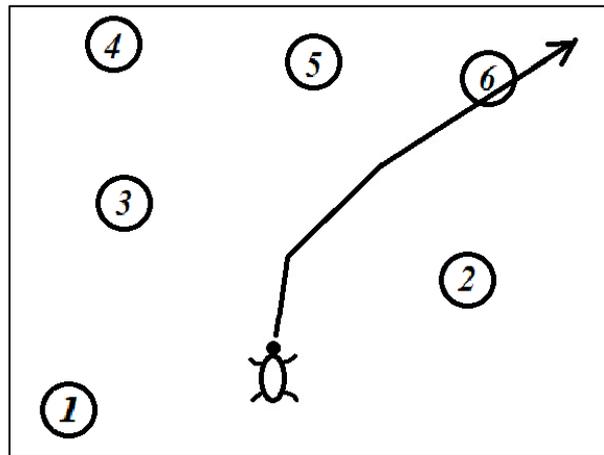


Рис. 2. Фрагмент маршрута робота

Тогда единицей описания точки в маршруте может служить тройка

$$p_n(C, Dir, Dist),$$

где  $n$  – номер шага, на котором робот видит объект с идентификатором  $C$ , расположенный относительно него по направлению  $Dir$  на расстоянии  $Dist$ . Причем на каждом шаге робот может видеть  $k$  точек ( $k \geq 1$ ), поэтому точка маршрута описывается множеством  $S_n = \{p_{n1}(C_1, Dir_1, Dist_1), \dots, p_{nk}(C_k, Dir_k, Dist_k)\}$ , а весь маршрут – множеством  $P = \{S_n\}$ .

С точки зрения определения относительного местоположения робота на плоскости значения имеют два параметра: направление и расстояние. Напомним, что фактически роботы могут совершать только четыре элементарных действия: двигаться вперед/ назад и совершать повороты. На основе этих действий очень сложно (практически невозможно) реализовать, например, сдвиг вправо или влево с приемлемой точностью. Поэтому было принято решение отказаться от учета расстояния и попробовать ориентироваться только по направлению.

Для работы с нечетким понятием «направление» были введены соответствующие лингвистические переменные, шкала пересчета направлений для области видимости в  $180^\circ$  приведена в Табл. 1.

Таблица 1. Шкала направлений

<i>Значение в градусах</i>	<i>Лингвистическая переменная</i>	<i>Обозначение</i>
0 – 22	Справа	R
23 – 67	Впереди справа	FR
68 – 122	Впереди	F
123 – 167	Впереди слева	FL
168 – 180	Слева	L

Теперь рассмотрим вопрос о том, каким должно быть поведение робота-последователя, которому нужно повторить этот маршрут. Считается, что любое достаточно сложное поведение складывается из совокупности простых поведенческих актов [17]. Поэтому имеет смысл разбить сложное поведение «пройти заданным маршрутом» на ряд более простых поведенческих процедур, например, «обойти объект X слева» или «двигаться к объекту Y». Кстати, опыты с муравьями показывают, что они ориентируются примерно так же: если изначально ориентир был левее маршрута и его сдвинули вправо, это заставляет муравьев прокладывать маршрут соответственно правее [11].

В качестве универсального средства для реализации этих процедур были использованы конечные автоматы [17]. Был создан набор автоматов Мили, каждый из которых реализует одну элементарную поведенческую процедуру, а именно: «обойти объект X справа / слева» (*BypassRigh / BypassLeft*), «двигаться к объекту X» (*MoveTo*), «обойти вокруг, поворачивая налево / направо» (*AroundL / AroundR*), «закончить путь» (*EndOfPath*) и автомат обхода препятствий (*Reflex*). Действиями робота-разведчика также управлял автомат, реализующий процедуру случайного блуждания (*Walk*). Схемы основных автоматов приведены на рис. 3-4. Пометка на дуге содержит условие перехода по этой дуге и действие, которое при этом выполняется.

На рис. 3,4 используются следующие обозначения:

$S$ ,  $T$  – начальное и конечное состояния автомата соответственно;

$N$  – количество тактов, в течение которых выполняется действие;

$IsLF$ ,  $IsRF$  – функции, возвращающие «истину», если робот левым (правым)

датчиком видит впереди препятствие;

*RandomTurn* – функция случайного выбора поворота направо или налево;

$L(X)$ ,  $FL(X)$ ,  $F(X)$ ,  $FR(X)$ ,  $R(X)$  – функции определения направления на объект  $X$ ;

$D(X)$  – функция определения расстояния до объекта  $X$ ;

*gofwd*, *goleft*, *goright*, *goback*, *stop* – действия робота (вперед, налево, направо, назад и стоп соответственно).

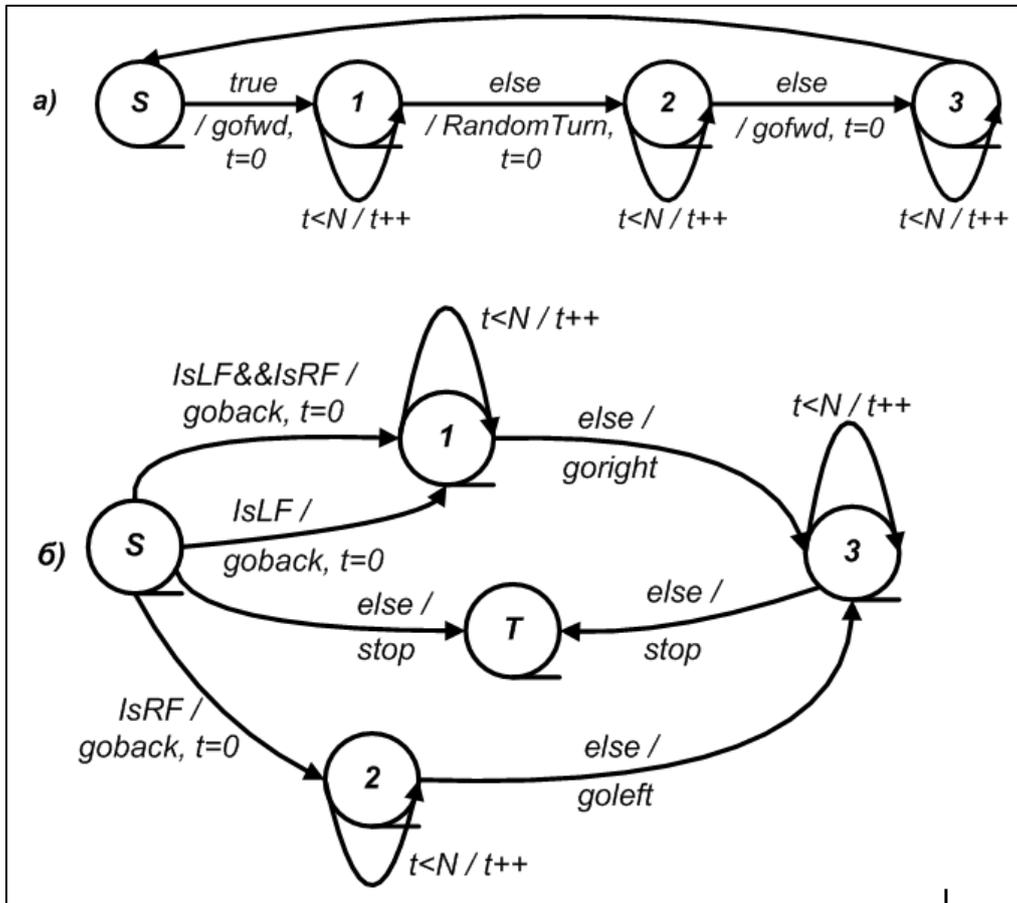


Рис. 3. Схемы автоматов, управляющих роботом-разведчиком:  
 а) случайное блуждание (Walk); б) обход препятствий (Reflex)

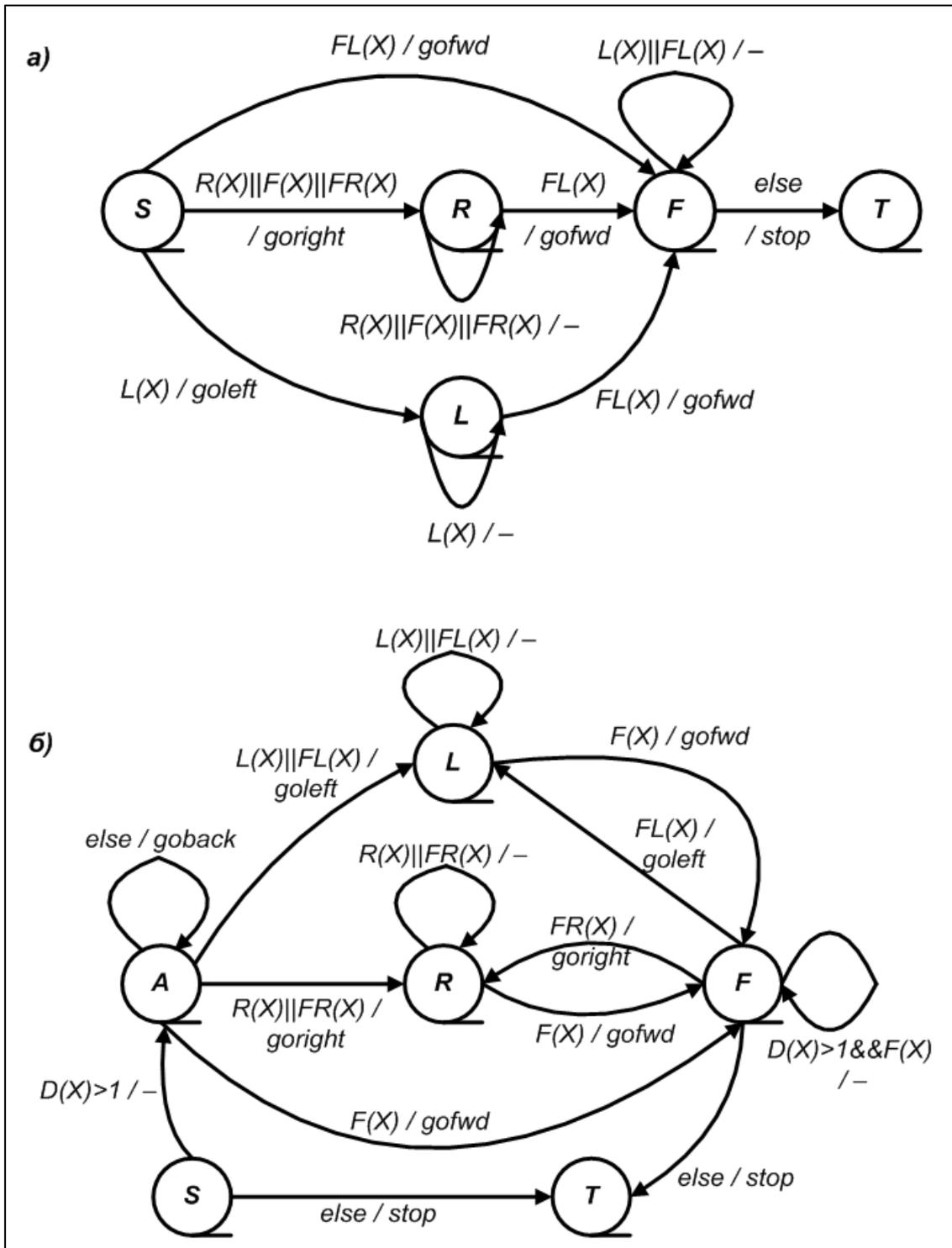


Рис. 4. Схемы автоматов, управляющих роботом-последователем:  
 а) обойти объект X справа (BypassRight); б) идти к объекту X (MoveTo)

Для того чтобы понять, как можно перейти от описания маршрута к действиям робота, сопоставим реальный путь разведчика с записью этого пути. Возьмем для примера фрагмент описания маршрута, где первым элементом каждой тройки указан номер шага, вторым – направление, третьим – цвет ориентира:

[1, впереди-справа, зеленый],  
 [2, впереди-справа, зеленый], [2, впереди-слева, желтый],  
 [3, слева, желтый], [3, справа, зеленый],  
 [4, впереди-справа, зеленый],  
 [5, впереди-слева, красный], [5, впереди-справа, зеленый],  
 [6, впереди-слева, красный], [6, справа, зеленый],  
 [7, слева, красный], [7, справа, зеленый], [7, впереди, синий],  
 [8, слева, красный], [8, впереди, синий],  
 [9, впереди, синий],...

Изучив реальный маршрут, соответствующий этому описанию (рис. 2), можно увидеть, что объект номер 2 («зеленый») робот обошел справа, объект номер 3 («желтый») – слева, объект номер 5 («красный») – справа, и движется по направлению к объекту номер 6 («синий»).

Для сопоставления описанием маршрута с самим маршрутом изменим форму записи описания, приведя его по точкам (по ориентирам):

**зеленый:** [1, впереди-справа], [2, впереди-справа], [3, справа],  
 [4, впереди-справа], [5, впереди-справа], [6, справа], [7, справа];  
**желтый:** [2, впереди-слева], [3, слева, желтый];  
**красный:** [5, впереди-слева], [6, впереди-слева], [7, слева], [8, слева];  
**синий:** [7, впереди], [8, впереди], [9, впереди]...

Из приведенного описания маршрута видно, что для области видимости в  $180^\circ$  можно выделить три устойчивых шаблона:

- 1) Если сначала робот увидел ориентир справа / впереди-справа и на последнем шаге, когда ориентир был виден, он был справа, то робот обошел его слева.
- 2) Если сначала робот увидел ориентир слева / впереди-слева и на последнем шаге он был слева, то робот обошел его справа.
- 3) Если сначала робот увидел ориентир в произвольном направлении, и на последнем шаге, когда ориентир был виден, он был впереди, впереди-справа или впереди-слева, то робот двигался к нему.

Таким образом, для определения способа обхода ориентира важно только направление, на котором находился ориентир перед тем, как пропасть из поля зрения робота. Это позволило сформировать набор правил интерпретации роботом-последователем маршрута, который приведен в Табл. 2. Правило,

отраженное в последней строке Табл. 2, также взято у муравьев: если муравей теряет ориентир, он начинает совершать круги вокруг того места, где по его ожиданиям должен находиться этот ориентир [11].

Таблица 2. Правила выбора действий в зависимости от направления на ориентир

<i>Направление на ориентир X</i>	<i>Действие</i>
Справа	Обойти объект X слева
Слева	Обойти объект X справа
Впереди, впереди-слева или впереди-справа	Двигаться к объекту X
Отсутствие ориентира	Обойти вокруг, поворачивая направо или налево (случайный выбор)

Для того чтобы такая интерпретация маршрута стала возможной, изменим форму его записи. Теперь единица описания точки маршрута выглядит так:

$$p^{\setminus}(n_k, Dir_k, C),$$

где  $n_k$  – номер шага, на котором ориентир с идентификатором  $C$  ушел из поля зрения, находясь по направлению  $Dir_k$ . Весь маршрут описывается множеством

$$P^{\setminus} = \{p_j^{\setminus}(n_{kj}, Dir_{kj}, C_j)\}, j=1, \dots, L,$$

где  $L$  – длина маршрута после преобразования.

Такое преобразование повлекло за собой существенное уменьшение количества хранимой информации и позволило, фактически, хранить маршрут в упакованном виде. Упаковка записи маршрута может осуществляться роботом-разведчиком «на ходу» по следующему алгоритму:

#### **Алгоритм создания описания маршрута**

```

Входные данные: данные локатора робота
Выходные данные: описание маршрута
RT = [] -- инициализация описания маршрута
cnt = 0 -- количество точек на маршруте
Пока (искомый объект не найден) цикл для робота-разведчика:
    Выполнение шага моделирования
    Получение очередной точки p[n, d, c] -- номер шага, направление, цвет
    Если (cnt==0): -- ориентиров еще нет, добавляем ориентир в описание
        cnt = cnt+1
        RT.добавить([ cnt, n, d, 0, 0, c])
    иначе:
        found = False -- признак наличия ориентира в описании маршрута

```

```

Цикл по i от 0 до (cnt - 1):  -- ищем ориентир такого же цвета
  Если RT[i][5]==p[2])      -- если тот же цвет и
    Если ((p[0] - RT[i][0])==1) -- ориентир был на предыдущем шаге
      RT[i][3] = p[0]        -- добавляем номер шага
      RT[i][6] = p[1]        -- и направление
      found = True           -- ориентир найден
    Конец если
  Конец если
Конец цикла по i
Если (not found): -- ориентир не найден
  cnt = cnt+1      -- считаем номер ориентира
  RT.добавить([ cnt, n, d, 0, 0, c]) -- добавляем ориентир в описание
Конец если
Конец если
Конец цикла моделирования

```

(Примечание: нумерация элементов массива начинается с 0).

В полученном описании маршрута робот-последователь должен отсортировать элементы описания в порядке, определяемом номером шага, на котором из поля зрения уходил очередной ориентир (ниже этот номер выделен полужирным шрифтом). После этого последовательность описания маршрута будет соответствовать последовательности изменения состояний робота, а ранее приведенное описание фрагмента маршрута примет следующий вид:

```

[2, впереди-слева, 3, слева, желтый],
[1, впереди-справа, 7, справа, зеленый],
[5, впереди-слева, 8, слева, красный],
[7, впереди, 9, впереди, синий]...

```

По этой записи видно, что робот часто одновременно проходит между двумя и более ориентирами. В связи с этим возникла проблема перехода от одной элементарной поведенческой процедуры к другой. Нельзя одновременно запускать несколько автоматов, управляющих движением робота, т.к. приходящие от них команды могут противоречить друг другу. Поэтому робот просматривает описание маршрута на несколько шагов вперед с тем, чтобы найти в нем видимые ему в данный момент ориентиры, и выбирает автомат в соответствии со следующим алгоритмом:

#### Алгоритм следования по маршруту

Входные данные: описание маршрута и данные локатора робота

```

Выходные данные: нет
CurrPoint = 0 -- текущий ориентир маршрута
Way[] – маршрут, LenWay – длина маршрута
Пока (искомый объект не найден) цикл для робота-последователя:
    Получить данные локатора.
    Сформировать список видимых ориентиров PList[], N – длина списка.
    Найти текущий ориентир маршрута ObjX в PList.
    Найти следующий ориентир маршрута ObjY в PList.
    Если ObjY найден,
        CurrPoint = CurrPoint+1
        Выбрать действие относительно ObjY
    иначе
        Если ObjX найден,
            то Выбрать действие относительно ObjX
        иначе -- робот не видит нужного ориентира
            found = False
            Цикл по j от 0 до N
                Цикл по i от CurrPoint до LenWay:
                    Если цвет(Way[i]) == цвет(PList[j]),
                        found = True
                        CurrPoint = i -- перейти на i-й ориентир
                        ObjX = Way[i]
                    Конец если
                Конец цикла по i
            Конец цикла по j
        Если (found),
            выбрать действие относительно ObjX
            иначе случайный выбор между (AroundL, AroundR)
        Конец если
    Конец если
Конец если
Выполнение шага моделирования
Конец цикла моделирования

```

Таким образом, если робот на текущем шаге не видит того ориентира, который должен видеть, он просматривает точки дальше по маршруту, сопоставляя их с тем, что видит. И, если находит в описании маршрута один из видимых ему ориентиров, ориентируется на него, «срезая» путь.

#### 4. РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТОВ

Для подтверждения работоспособности предложенного метода была проведена серия экспериментов по имитационному моделированию.

Эксперименты проводились на упрощенной задаче фуражирования: не рассматривался этап возвращения робота-разведчика в «дом» и процесс передачи сведений о маршруте. При использовании данного метода возвращение в точку отправления может проводиться по тому же маршруту, только точки в описании маршрута надо отсортировать в обратном порядке и поменять местами процедуры «обход слева» и «обход справа».

Эксперимент заключался в том, что сначала робот-разведчик перемещался по полигону случайным образом с целью найти объект заданного цвета, не имея заранее данных о его местоположении. После нахождения объекта сформированный маршрут передавался второму роботу, который должен был пройти по этому маршруту. Моделирование проводилось с использованием системы многоагентного моделирования Kvoqum, созданной в НИЦ «Курчатовский институт» и работающей под управлением фреймворка ROS (Robot Operating System) [18].

Роботы перемещались по полигону – области  $100 \times 100$  клеток. В произвольном месте полигона располагается «дом» – место, с которого начинался любой маршрут (рис.5). В разных сериях экспериментов полигон представлял собой ограниченную область или тороидальную поверхность с препятствиями или без них. Время моделирования составляло 1000 тактов. Ориентиры располагались как регулярным, так и произвольным образом, но так, чтобы в каждой точке полигона существовал хотя бы один ориентир, «видимый» из этой точки. Зона видимости определялась расстоянием, на котором агент способен «увидеть» объект (20 клеток).

Как видно из рис.5, на котором приведены примеры результатов моделирования, робот-последователь может довольно точно повторять маршрут робота-разведчика (рис.5,а), иногда «срезает» путь (рис.5,в-г), а иногда сходит с маршрута, но потом возвращается на него (рис.5,б).

На рис.5,в боковые ответвления маршрута робота-разведчика – это его попытка преодолеть препятствие. Когда робот сталкивается с препятствием одним или двумя датчиками препятствий, управление передается автомату

*Reflex*, который заставляет робота отойти назад и повернуть направо или налево. Как видно из рис.5,в, робот-последователь не повторяет эти маневры, и таким образом сокращает свой путь.

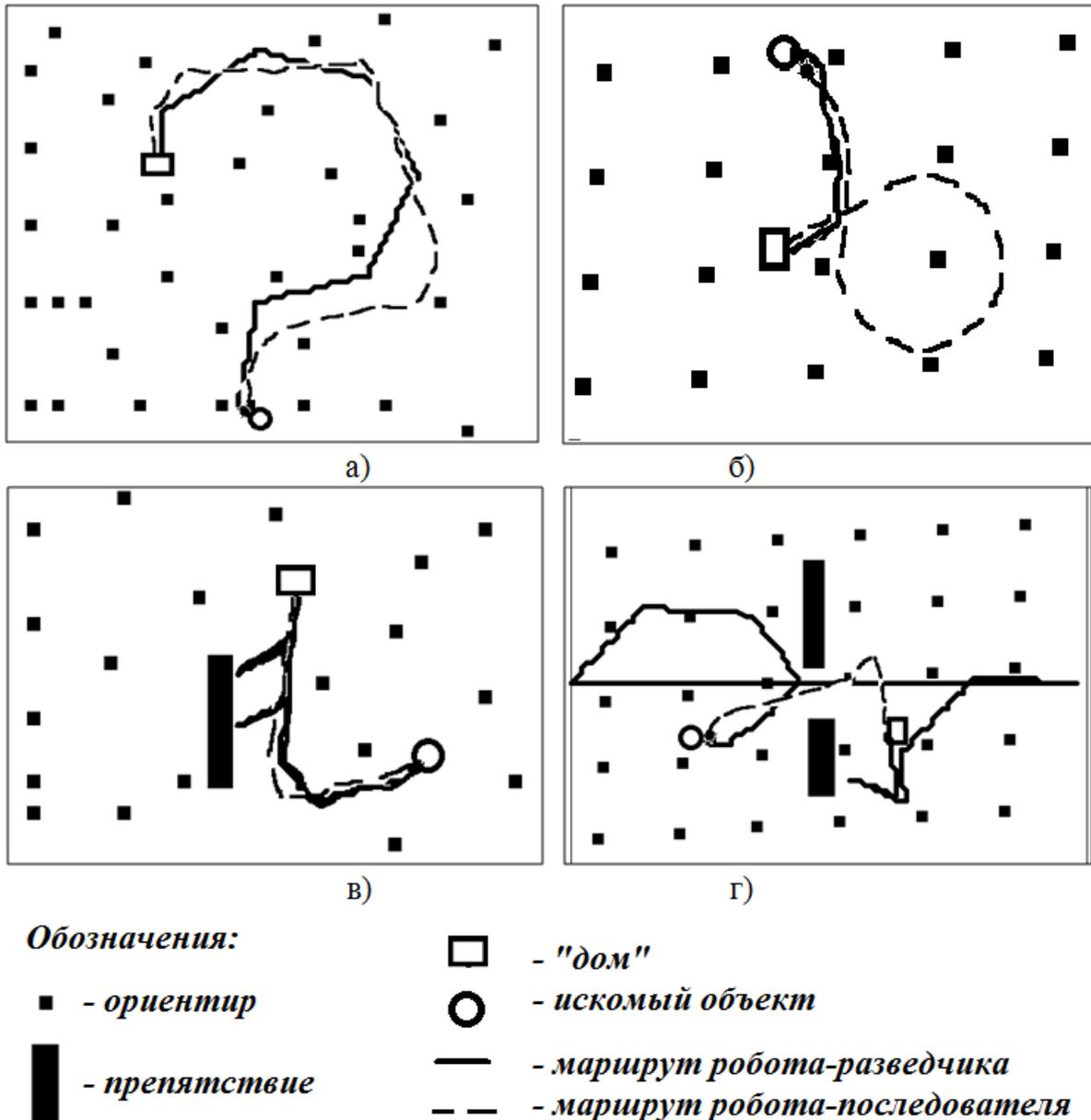


Рис. 5. Примеры маршрутов робота-разведчика и робота-последователя

На рис.5,в боковые ответвления маршрута робота-разведчика – это его попытка преодолеть препятствие. Когда робот сталкивается с препятствием одним или двумя датчиками препятствий, управление передается автомату *Reflex*, который заставляет робота отойти назад и повернуть направо или налево. Как видно из рис.5,в, робот-последователь не повторяет эти маневры, и таким образом сокращает свой путь.

В Табл.3 приведены обобщенные результаты моделирования, а усредненные результаты моделирования приведены в Приложении (Табл. 4). Из Табл.3 видно, что приведенный маршрут  $P'$  на порядок короче исходного  $P$ . На длинных маршрутах (более 250 шагов) робот-последователь часто «срезает» путь, переходя на следующий ориентир, и поэтому делает меньше шагов, чем робот-разведчик (рис. 5,г). А на коротких маршрутах (до 150 шагов) наоборот робот в среднем добирается до цели за большее число шагов, чем робот-разведчик. Это объясняется тем, что робот-последователь иногда «теряется», он вынужден поворачиваться вокруг себя, чтобы снова увидеть нужный ориентир (рис. 5,б), а «срезать» на коротком маршруте нечего.

Таблица 3. Соотношение путей и количества шагов роботов

Препятствия	Ориентиры	Маршрут	Длина упакованного маршрута, %	Изменение количества шагов, %	Кол-во неудач, %
Нет	Регулярные	Короткий	8,7	127	7,9
		Средний	9,2	125	2,4
		Длинный	8,9	97	2,5
	Нерегулярные	Короткий	8,8	104	3,3
		Средний	8,7	107	3,0
		Длинный	8,7	99	3,5
Есть	Регулярные	Короткий	8,8	110	8
		Средний	8,6	93	6,3
		Длинный	9,2	75	6,3
	Нерегулярные	Короткий	10,0	84	7,1
		Средний	9,5	81	3,0
		Длинный	10,0	66	3,4

Неудачи в основном обусловлены тем, что робот-последователь, соблюдая только общее направление, иногда слишком сильно отклоняется в сторону от маршрута робота-разведчика, выходит из зоны видимости следующего ориентира и не успевает вернуться на маршрут за оставшееся время. Кроме того, при наличии препятствий они могут преградить роботу путь к ориентире, и ему не хватает времени на обход препятствия и возвращения на маршрут.

## 5. ЗАКЛЮЧЕНИЕ

В дальнейшем планируется применять разработанный метод в задачах фуражирования, патрулирования территории и рекогносцировки для ориентации реальных роботов.

Предлагаемый метод имеет два принципиальных ограничения. Во-первых, маршруты должны начинаться из одной из той же точки и роботы должны быть повернуты в одну сторону. Во-вторых, ориентиры должны быть расположены таким образом, чтобы из каждой точки полигона робот мог увидеть хотя бы один ориентир. Другие параметры, такие как регулярность и нерегулярность ориентиров, расстояние до цели, наличие препятствий не влияют существенно на успешное прохождение маршрута роботом-последователем.

Для решения данной задачи нам потребовалось осуществить комплексирование информации, поступающей на датчики (сенсоры) робота. Полученная таким образом структура может быть соотнесена с информационной гранулой [19], которая лежит в основе построения обобщенного представления динамических пространственных отношений. В определенном смысле обладающий такими возможностями робот является когнитивным агентом, чья когнитивная функция обеспечивает процессы познания роботом внешнего мира.

Использование конечных автоматов для выполнения поведенческих процедур также имеет свои преимущества как универсальное средство реализации управления [20]. В принципе, можно создать автомат верхнего уровня управления, который будет осуществлять запуск поведенческих процедур в зависимости от текущей ситуации. Таким образом, использование явного автоматного представления процедур поведения агентов позволяет говорить о реализации единой, монолитной системы управления, основанной на нейросетевой парадигме. Это обусловлено эквивалентностью конечных автоматов и рекуррентных нейронных сетей.

## Библиографический список

1. Карпов В. Э. Модели социального поведения в групповой робототехнике. / Управление большими системами. Выпуск 59. – М.: ИПУ РАН, 2016. – с.165-232.
2. Кулинич А.А. Модель командного поведения агентов (роботов): когнитивный подход. // Управление большими системами. М.: ИПУ РАН, вып. 51, 2014. – с.174-196.
3. Thrun, S. Simultaneous localization and mapping (Review) / Springer Tracts in Advanced Robotics, Volume 38, 2008, Pages 13-41.
4. Moller, R., Vardy, A. Local visual homing by matched-filter descent in image distances / (2006) Biological Cybernetics, 95 (5), Pages 413-430. doi: 10.1007/s00422-006-0095-3
5. Baddeley, B., Graham, P., Husbands, P., Philippides, A. A model of ant route navigation driven by scene familiarity. / PLoS Computational Biology, Volume 8, Issue 1, January 2012, Article number e1002336.
6. Kim, P., Kurabayashi, D. Efficient formation of pheromone potential field by filtering interaction. / Journal of Robotics and Mechatronics, Volume 24, Issue 4, 2012, Pages 561-567.
7. Hoff, N., Wood, R., Nagpal, R. Distributed colony-level algorithm switching for robot swarm foraging. In: Springer Distributed Autonomous Robotic Systems, 2013, pp. 417-430.
8. Lee, J.-H., Ahn, C.W., An, J., 2013. A honey bee swarm-inspired cooperation algorithm for foraging swarm robots: an empirical analysis. In: 2013 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM), IEEE, Novotel, Wollongong, Australia, Pages 489-493.
9. Meng, Z., Zou, B., Zeng, Y. Considering direct interaction of artificial ant colony foraging simulation and animation. / J. Exp. Theor. Artif. Intell., 24 (1) (2012), Pages 95-107.
10. Zedadraa O., Seridib H., Jouandeauc N., Fortinod G. A Cooperative Switching Algorithm for Multi-Agent Foraging / Engineering Applications of Artificial

Intelligence, Volume 50, April 2016, Pages 302–319.

11. Длусский Г. М. Муравьи рода Формика. – Москва: Наука, 1967. – 236 с.
12. Захаров А.А. Муравей, семья, колония. – М.: Наука, 1978. – 144 с.
13. Калущая А.П., Тарасов В.Б. Моделирование взаимодействия робота с внешней средой на основе пространственных логик и распространения ограничений. / Программные продукты и системы, 2010, №2. – с.174-178.
14. Карпов В.Э. Частные механизмы лидерства и самосознания в групповой робототехнике // XIII национальная конференция по искусственному интеллекту с международным участием КИИ-2012 (16-20 октября 2012 г., Белгород): Труды конференции в 4-х томах. Т.3. – Изд-во БГТУ, 2012. – 309 с. – с. 275-283.
15. Muller M., Wehner R. Path integration in desert ants, *Cataglyphis fortis*. / Proceedings of the National Academy of Sciences of USA, 85: 5287-5290.
16. Wehner R. The architecture of the desert ant's navigational toolkit (Hymenoptera: Formicidae). Myrmecol News 12 Vienna, September 2009, Pages 85-96.
17. Цетлин М.Л. Исследования по теории автоматов и моделированию биологических систем. – М.: Наука, 1969. – 316 с.
18. Robot Operating System. – <http://www.willowgarage.com/pages/software/ros-platform>
19. Тарасов В.Б. Гранулярные структуры измерений в интеллектуальных средах: васильевские и белнаповские сенсоры и модели их взаимодействия. / Информационно-измерительные и управляющие системы. 2013. Т. 11. № 5. – с. 65-74.
20. Karpov V.E. Emotions and Temperament of Robots: Behavioral Aspects. Journal of Computer and Systems Sciences International, 2014, Vol. 53, No. 5, Pages 743–760.

**Concerning Presentation of a Route for a Mobile Robot Based on Visual Guides**

**I.P.Karpova**, karpova\_ip@mail.ru,

National Research University Higher School of Economics (HSE), Moscow, 101000, Russian Federation,

Moscow Institute of Physics and Technology, Dolgoprudny, 141701, Russian Federation

*Corresponding author:*

**Karpova I.P.**, Ph.D., Associate Professor, National Research University, Higher School of Economics (HSE), Moscow, 101000, Russian Federation,  
Moscow Institute of Physics and Technology, Dolgoprudny, 141701, Russian Federation

e-mail: karpova\_ip@mail.ru

*Received on July 29, 2016*

*Accepted on August 15, 2016*

*One of the widespread approaches to the issues of control in the group robotics is application of the social behavior models in the groups of robots. In this paper the author proposes to use this approach to fulfill the tasks of foraging. As a role model a Formicidae ant is proposed. This task is considered as a combination of three stages: finding food, returning to the ant hill and repeating the way to the place where food was found. It was proven that in order to come back home and repeatedly walk the way the Formicidae ants were navigated predominantly by the visual means using vector navigation (path integration) and landmark-guidance mechanisms. The basis of the proposed method is formed by the principles of memorizing the way by the visual landmarks and fuzzy control. The model of describing the way is introduced to the robot, which can define colors of the landmarks and approximately sense the direction to the landmark in respect to itself. A pattern for formation of a succinct way description was created, with the help of which the scout robot memorizes the way to the “food”. Certain regulations were developed, which let the follower robot transfer from the description of the route to the actions of its reproduction and in many ways copy an ant’s behavior. The actions are presented as elementary behavioral procedures, and each behavioral procedure is realized as a finite state automata. The results of the simulation modeling, which was conducted with the help of the framework of ROS based modeling system, are presented. Experiments were conducted in polygons with barriers and without them, with regular and irregular placing of the landmarks. As a quality criterion for the proposed method the author offers to consider a successful passing of the route by the follower robot, and this indicator in different series of experiments varies from 92 up to 98 %. The proposed method does not require robot’s great computation capacity and advanced sensory abilities. The developed method can also be applied to the tasks of reconnaissance and patrolling.*

**Keywords:** *collective robotics, cognitive agent, spatial relations, social behavior models, granulation of information, foraging task.*

**Acknowledgements:** This work was supported by RFBR grant 16-11-00018.

For citation:

**Karpova I. P.** Concerning Presentation of a Route for a Mobile Robot Based on Visual Guides, *Mekhatronika, Avtomatizatsiya, Upravlenie*, 2017, vol. 18, no. 2, pp.81-89. DOI: 10.17587/mau.18.81-89