
СЕКЦИЯ 8 | ИНТЕЛЛЕКТУАЛЬНЫЕ РОБОТЫ

УДК 681.5+004.8

ПРОЦЕДУРА ГОЛОСОВАНИЯ В ОДНОРОДНЫХ КОЛЛЕКТИВАХ РОБОТОВ¹

В.Э. Карпов (*vkarpov@hse.ru*)
МИЭМ НИУ ВШЭ, Москва

В работе предложены решения задач определения лидера и распределения ролей в однородной группе роботов. Показано, что используя исключительно локальное взаимодействие, возможен переход от роя к коллективу роботов с иерархической организацией. В основе процедуры выбора лидера лежит алгоритм локального переголосования, а распределение ролей может быть осуществлено волновым методом.

Введение

Активные исследования в области создания систем взаимодействующих роботов ведутся уже почти четверть века. Такие направления, как коллективная, роевая, стайная и пр. робототехника, заняли прочные позиции в современной робототехнике и теории многоагентных систем. Однако до сих пор подавляющее число исследований в этой области остается на теоретическом, модельном уровне. Это хорошо видно по многочисленным обзорам, например, таким, как [Zhiguo et al., 2012] и [Yogeswaran & Ponnambalam, 2010]. Среди множества задач роевой робототехники, до сих пор остающимися недостаточно полно раскрытыми, выделим две – за-

¹ Работа выполнена при поддержке РФФИ (проект № 14-01-00817)

дачу определения лидера в группе роботов, а также задачу распределения ролей среди членов этой группы.

Лидерство. Одной из принципиальных особенностей роевой робототехники является локальный характер взаимодействия роботов друг с другом, а также роботов со средой [Zhiguo et al., 2012]. Такое взаимодействие называется неявной коммуникацией (implicit communication) [Yogeswaran & Ponnambalam, 2010]. Речь идет о том, что каждый робот группы непосредственно взаимодействует лишь со своими соседями, находящимися в некоторой ограниченной зоне видимости. Отсюда обычно следует, что в такой системе роботы самостоятельно принимают решения о дальнейших действиях, опираясь на некоторые простые правила локального взаимодействия.

Вместе с тем, однако, подавляющее число примеров решения задач в области роевой робототехники касается согласованного движения роя (очевидная, наглядная и достаточно простая задача, которая в [Zhiguo et al., 2012] выделена как метод "Следование за лидером" – "Leader-Follower method"). При этом считается, что в группе имеется априори заданный лидер, который и задает это движение. Правила локального взаимодействия могут быть самыми разными. От сугубо формальных [Павловский и др., 2010] до весьма экзотических. Например, в [Dewi et al., 2012] правила стайного движения основаны на модели "пружин и амортизаторов", определяющей притяжение и отталкивание особей к лидеру. В [Карпов, 2012], напротив, описан сугубо технический прием, позволяющий определить лидера для решения задачи согласованного движения. Иной подход к определению лидера предложен в [Даринцев, 2006], где описывается динамическое выделение агентов-координаторов (лидеров), основанное на "геометрических" построениях – лидерами групп становятся роботы, ближе всех расположенные к месту проведения неких технологических действий.

В любом случае мы имеем дело либо с априори заданным лидером, либо с неким сугубо техническим приемом, определяющим лидера в условиях некоторой конкретной задачи. Нас же интересует проблема выявления лидера в более общем случае, с точки зрения механизмов локального информационного обмена.

Распределение задач. Для решения задачи согласованного движения достаточно наличия лидера. Однако для более сложных

задач, решаемых группой роботов, требуется дифференциация их функций и, вообще говоря, распределение задач между роботами. Пожалуй, это одно из наиболее проблемных мест роевой робототехники. В уже упомянутых обзорах распределение задач в коллективе роботов рассматривается лишь декларативно. В лучшем случае упоминаются некоторые физические модели, методы распределенного планирования, оптимизации и пр. общие механизмы ([Каляев и др., 2009]). Причиной этому является то, что дифференциация функций и распределение задач не рассматривается роевой робототехникой как актуальная проблема, полагая, что рой должен решать лишь простые, массовые задачи типа согласованного движения. Это, безусловно, снижает значимость роевого подхода и, вообще говоря, идет в разрез с основным декларируемым тезисом роевой робототехники (роевая робототехника, как способ решения сложных задач совокупностью простых технических устройств – роботами).

Мы же будем полагать, что и задача формирования лидера, и задача распределения ролей крайне важны для развития роевой робототехники. Рассмотрим варианты решения этих задач на примере такой модели организации группы роботов, как статический рой.

Статический рой

Статический рой характеризуется отсутствием заданного управляющего центра и представляет собой некую фиксированную в данный момент времени сеть – совокупность агентов [Карпов, 2013]. Основные свойства статического роя – это активность, локальность взаимодействия и функциональная неоднородность.

Важным вопросом представляется организация механизма такой функциональной неоднородности. Рассмотрим следующую задачу. Пусть имеется множество агентов (роботов), способных к локальному информационному обмену между ближайшими соседями. Далее, в некоторый момент времени статический рой должен реализовать некую процедуру распределения ролей: кто-то должен стать управляющим центром, кто-то – выполнять функции обработки информации, кто-то – сбора информации из внешней среды и т.д.

Общие соображения относительно принципа распределения ролей могут базироваться на следующих очевидных рассуждениях:

узел сети (агент), имеющий максимальное количество связей, становится претендентом на роль управляющего центра. Его ближайшее окружение – анализаторы информации, подготавливающие ее для принятия решения. Узлы, расположенные на периферии сети, отвечают за сбор информации. На Рис. 1 изображен пример сети.

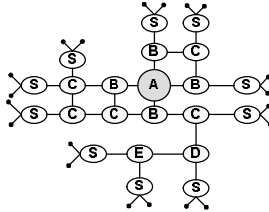


Рис. 1. Пример организации сети

Здесь узел *A* становится управляющим центром, его ближайшие соседи (*B*) – анализаторами, а периферийные узлы (*S*) будут отвечать за внешнюю сенсорику. При этом центральным вопросом является то, каким образом узлы-агенты выберут центральный, главный узел. Итак, рассмотрим далее возможные способы организации такого голосования.

Задача голосования

Сформулируем задачу следующим образом. Пусть имеется множество агентов с ограниченными коммуникационными возможностями. Это означает, что агенты способны лишь к непосредственному локальному взаимодействию между соседями. Для этого можно предположить, что агенты имеют фиксированное число коммуникационных портов – точек контакта, позволяющим им устанавливать каналы обмена информацией. Задача состоит в том, чтобы агенты выбрали единственного лидера путем голосования.

Пусть в некоторый момент времени агенты получают глобальный сигнал о начале голосования. В этот момент времени каждый агент устанавливает каналы связи со своими соседями. Таким образом, образуется некий в общем случае направленный граф. Вершинами его являются агенты, а входящие дуги интерпретируются как возможность получения информации от узла-источника – образуется канал связи. Зафиксируем статический рой, т.е. будем полагать, что далее его топология меняться не будет. Каждый агент описывается четверкой

$$A = (\alpha, L, C, W),$$

где α – идентификатор или имя агента, L – список агентов-соседей, от которых агент может получать информацию (входящие дуги), C – идентификатор "кандидата", за которого голосует агент α , W – вес кандидата C , т.е. число голосов, которое, по мнению агента, следует отдать за кандидата.

Суть процедуры голосования заключается в том, что каждый агент определяет, за кого голосуют его соседи. При этом, в зависимости от веса кандидата, за которого голосует сосед, агент может поменять свое мнение и проголосовать за того же кандидата, что и его сосед.

На Рис. 2 представлен один такт такой схемы голосования. Пометки вершин означают следующее: в "числителе" указывается идентификатор агента α , а величины C_α и W_α означают идентификатор кандидата и его вес соответственно.

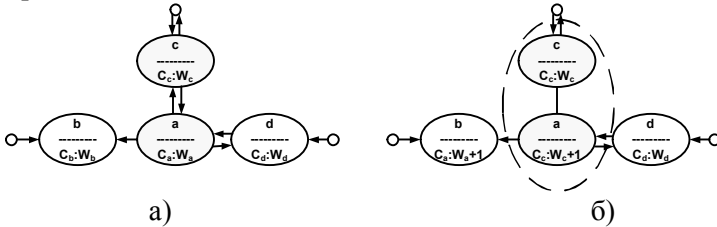


Рис. 2. Один такт голосования: а – начальное, б – конечное распределение голосов

Предположим, что агент a голосует за кандидата C_a , а агент c – за кандидата C_c . Если вес W_a окажется меньше веса W_c , то агент a может поменять свое мнение и переголосовать. При этом к весу нового кандидата будет добавлен еще один голос. Вероятность того, что агент i изменит свое мнение под влиянием мнения агента j (оппонента), может быть определена так:

$$P_{ij} = \frac{W_i}{W_i + W_j}$$

Т.е. склонность к перемене своего мнения естественным образом зависит от степени убежденности (или веса) кандидата. Распределение голосов кандидатов и их весов в начальный момент времени осуществляется также вполне естественно: каждый агент голосует за себя (объявляет себя кандидатом), а вес этого решения

равен количеству соседей этого агента. Ниже приведен алгоритм поведения агента при голосовании.

Алгоритм G1(α). Принятие решения агентом

α – агент

C_α – "кандидат", за которого голосует агент α

W_α – вес кандидата

L_α – список агентов-соседей

procedure G1 (α)

1. Выбрать среди своих соседей оппонента A_{op} :

$$A_{op} \in L_\alpha, C_{op} \neq C_\alpha, W_{op} = \max_{i \in L_\alpha} W_i$$

2. Вычислить значение вероятности изменения мнения

$$p_\alpha = \frac{W_{op}}{W_\alpha + W_{op}}$$

3. С вероятностью p_α изменить мнение: $C_\alpha \leftarrow C_{op}$
 $W_\alpha \leftarrow W_{op} + 1$

end procedure

Можно "заглубить" алгоритм принятия решения, заставляя агента безусловно менять свое мнение о кандидате, если в его окружении имеется более сильный "оппонент". Если же наблюдается паритет между весами мнений агента и его самого сильного оппонента, то здесь выбор решения может быть осуществлен уже вероятностным образом. Общая схема голосования может выглядеть так:

Алгоритм V(A). Голосование

A – множество агентов

α – агент

C_α – "кандидат", за которого голосует агент α

W_α – вес кандидата

L_α – список агентов-соседей

eof – флаг завершения процедуры голосования

procedure V (A)

1. $eof \leftarrow false$

2. **for all** $\alpha \in A$ **do** -- Инициализация агентов

3. $C_\alpha \leftarrow \alpha$

$W_\alpha \leftarrow \dim(L_\alpha)$

4. **end for**

```

5. while not eof do           -- Основной цикл голосования
6.   for all  $\alpha \in A$  do -- Цикл по всем агентам
7.     G1( $\alpha$ )
8.   end for
9.   «Определение условия завершения процедуры голосо-
сования eof»
10. end while
end procedure

```

В этом алгоритме наиболее проблемным является п.9: «Определение условия завершения процедуры голосования *eof*». В отсутствие глобальной информации о состоянии сети, агент должен сам принимать решение о том, что голосование закончено. Получаемая от ближайшего окружения информация является явно не достаточной для этого, поэтому возможны два варианта поведения агента:

1. Считать, что голосование должно быть завершено максимум через некоторое определенное количество тактов. Для этого необходима верхняя оценка количества шагов алгоритма голосования.

2. Реализовать некоторую процедуру обмена сообщениями, которая могла бы определить, что голосование завершено, и никто из агентов уже не меняет свое решение.

Ниже приведены примеры работы процедуры голосования. На Рис. 3а изображены 3 такта голосования для плотной группы роботов.

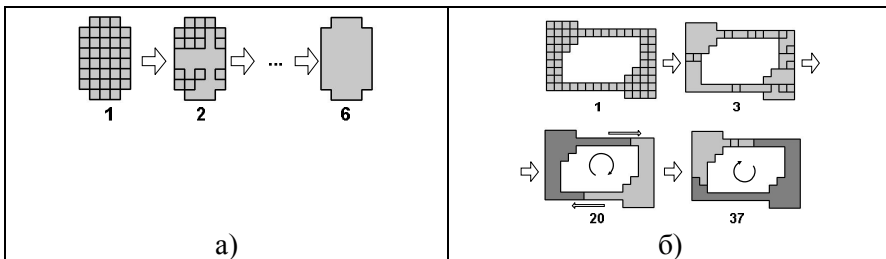


Рис. 3. а - процедура голосования в плотной группе (такты 1, 2 и 6), б - Циклическая процедура голосования (такты 1, 3, 20 и 37)

На первом такте каждый агент голосует за себя, поэтому количество клеток, обозначающих "границы" распределения голосов

за соответствующих кандидатов, равно числу роботов. На втором такте в результате переголосования происходит укрупнение областей, голосующих за выбранных кандидатов. Наконец, на 6-м такте все голоса отданы за одного кандидата и процедура голосования завершается.

На Рис. 3б приведен пример процесса голосования в крайне неблагоприятных условиях. Здесь наблюдаются две явно выраженные зоны, соединенные двумя перешейками. В такой ситуации наблюдается циклический процесс распространения голосов. К 20-му такту голосования образованы две устойчивые области, каждая из которых голосующая за своего кандидата. И далее начинается процесс циклического переголосования. Это хорошо видно на рисунке, соответствующему 37 такту, когда предпочтения областей, фактически, поменялись местами. Процесс устанавливается лишь к такту 51, когда в системе прекращаются колебания и остается единственный кандидат в лидеры.

Судя по результатам имитационного моделирования, процесс голосования сходится. При этом количество тактов голосования не превышало числа роботов в группе.

Распределение задач. При отсутствии морфологических различий между агентами, распределение ролей в статическом рое определяется исключительно текущей топологией системы. Сам процесс распределения представляет собой известную процедуру распространения волны управления. Инициатором распространения является лидер R_0 . Непосредственные соседи лидера ("ближний круг") получают от него иницирующий пакет, согласно которому им назначается роль R_1 и т.д. Таким образом, роль i -го робота определяется ролями его окружения:

$$R_i = \max_{k \in Z} R_k + 1$$

Волновое распределение ролей реализуется исключительно локальным взаимодействием.

Заключение

Итак, были предложены простые и эффективные механизмы решения таких важных задач роевой робототехники, как определение лидера и распределение ролей между членами группы. Под эффективностью понимается их приемлемость для роботов с ограничен-

ными когнитивными возможностями (ограниченность сенсорики, вычислительных мощностей, каналов связи и т.д., т.е. всего того, что характерно для роевой робототехники). Несмотря на свою простоту, реализация этих механизмов позволяет говорить о наличии принципиальной возможности образования действительно сложных по своей организации структур в однородных коллективах.

Список литературы

- [Даринцев, 2006] Даринцев О.В. Система управления коллективом микророботов // "Штучный интеллект" 4, 2006, №4, с.391-399.
- [Иванов, 2011] Иванов Д.Я. Использование принципов роевого интеллекта для управления целенаправленным поведением массово применяемых микророботов в экстремальных условиях // Изв. высших учебных заведений. М.:Машиностроение. 2011, №9, с.70-78.
- [Каляев и др., 2009] Каляев И.А., Гайдук А.Р., Капустян С.Г. Модели и алгоритмы коллективного управления в группах роботов. М.: Физматлит, 2009. 280с.
- [Карпов, 2012] Карпов В. Э. Частные механизмы лидерства и самосознания в групповой робототехнике // Тринадцатая национальная конференция по искусственному интеллекту с международным участием КИИ-2012. Т. 3. Белгород: Белгородский государственный технологический университет им. В.Г. Шухова, 2012. с. 275-283.
- [Карпов, 2013] Карпов В.Э. Управление в статических роях. Постановка задачи //Тр. VII-й Международной научно-практической конференции Интегрированные модели и мягкие вычисления в искусственном интеллекте, Коломна. Т.2. М.: Физматлит, 2013. с.730-739.
- [Павловский и др., 2010] Павловский В.Е., Кирикова Е.П., Павловский В.В. Моделирование поведения больших групп роботов в среде с препятствиями //Тр. научно-технического семинара "Управление в распределенных сетцентрических и мультиагентных системах". СПб.: ОАО "Концерн ЦНИИ «Электроприбор» ", 2010. с.10-13.
- [Dewi et al., 2012] Dewi T., Risma P., Oktarina Y. Wedge Formation Control of Swarm Robots. //The 14th Industrial Electronics Seminar, Electronic Engineering Polytechnic Institute of Surabaya (EEPIS), Indonesia, 2012. P.294-298
- [Yogeswaran & Ponnambalam, 2010] Yogeswaran M. and Ponnambalam S. G. (2010). Swarm Robotics: An Extensive Research Review, Advanced Knowledge Application in Practice, InTech. P.259-278.
- [Zhiguo et al., 2012] Zhiguo Shi, Jun Tu, Qiao Zhang, Lei Liu, Junming Wei, A Survey of Swarm Robotics System //Proc. of the Third Intern. Conf. on Advances in Swarm Intelligence, Shenzhen, China, 2012. V.1, P.564-572.