



МОДЕЛЬ НЕЧЕТКОГО РЕГУЛЯТОРА ДЛЯ УПРАВЛЕНИЯ ДВИЖЕНИЕМ МОБИЛЬНОГО РОБОТА

MODEL OF FUZZY CONTROLLER FOR CONTROLLING THE MOVEMENT OF A MOBILE ROBOT

УДК 681.5.017

ВИНОГРАДОВ АРТЕМ НИКОЛАЕВИЧ

vinogr-artem@yandex.ru

ТЕРЕНТЬЕВ АЛЕКСЕЙ ИГОРЕВИЧ

aaattt@mail.ru

ПЕТРОВ ОЛЕГ ВЛАДИМИРОВИЧ

Национальный исследовательский университет «МИЭТ»
124498, г. Москва, г. Зеленоград, площадь Шокина, 1

Построена модель системы управления движением мобильного робота на основе нечеткой логики. Выбраны входные и выходные переменные, предложены функции принадлежности. Проведено сравнение функционирования классического и нечеткого регуляторов.

Ключевые слова: нечеткая логика; алгоритм Мамдани; мобильный робот; LabView.

The paper presents a model of motion control of a mobile robot based on fuzzy logic, as well as its input/output variables and membership functions. A comparison with a model built on the basis of a classical regulator has been made.

Keywords: fuzzy logic; Mamdani algorithm; mobile robot; LabView.

Одной из проблем управления мобильными роботами является построение траектории движения. Эта задача является одной из важных, поскольку построение оптимальной траектории движения позволяет повысить эффективность функционирования мобильного робота. Следует отметить, что эта задача может быть решена с использованием системы управления на основе классического регулятора [1]. Однако получаемые результаты не всегда обеспечивают высокую эффективность функционирования робота.

В то же время, в условиях автономного управления (отсутствие связи с оператором) робот не может получать корректирующее управление извне. Таким образом, задача должна решаться от начала и до конца непосредственно системой управления робота. Это решение должно происходить в условиях определенных ограничений на техническую составляющую (компактность, низкое энергопотребление), а это сказывается на используемом алгоритмическом обеспечении.

Объектом управления в данной работе является роботизированное шасси. Система управления шасси состоит из электродвигателей и приводов, а также вычислительного устройства. Обязательной частью мобильного робота являются датчики. В данной работе будет моделироваться работа навигационной системы, отвечающей за определение ориентации робота в пространстве. В модели учитывается, что навигационная система работает с погрешностью.

Рассмотрим решение задачи управления движением мобильного робота из точки А с координатами (x_n, y_n, α_n) в точку В с координатами $(x_{n+1}, y_{n+1}, \alpha_{n+1})$, где x — координата робота по оси абсцисс; y — координата робота по оси ординат; α — угол ориентации робота (рис. 1). Моделируется движение робота в двухмерном

VINOGRADOV ARTEM N.

vinogr-artem@yandex.ru

TERENTYEV ALEXEY I.

aaattt@mail.ru

PETROV OLEG V.

National Research University of Electronic Technology
1 Shokin Square, Zelenograd, Moscow, 124498, Russia

пространстве без препятствий в режиме реального времени. Координаты робота определяются в декартовой системе координат, выбираемой из соображений удобства при дальнейшем отображении траектории. Особенностью решаемой задачи является учет ориентации робота в точке назначения. Под углом ориентации робота понимается разница между вектором движения робота и осью абсцисс в выбранной системе координат. Траектория движения будет строиться в зависимости от того, в каком

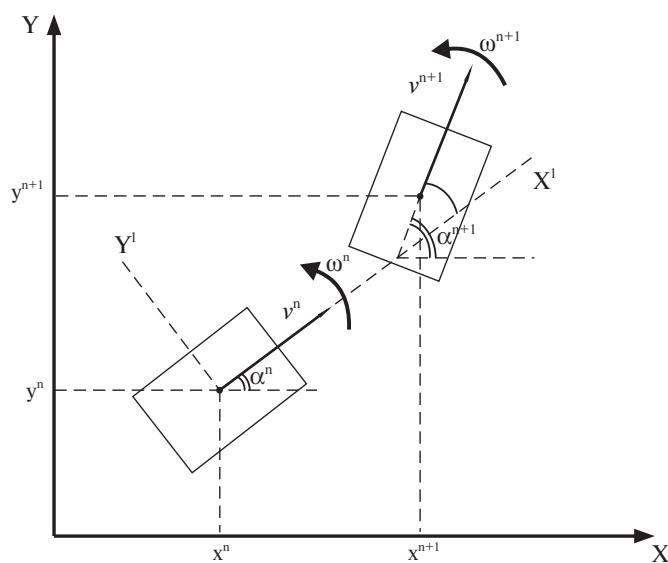


Рис. 1. Иллюстрация задачи управления движением мобильного робота
Fig. 1. An illustration of the task of controlling the movement of a mobile robot

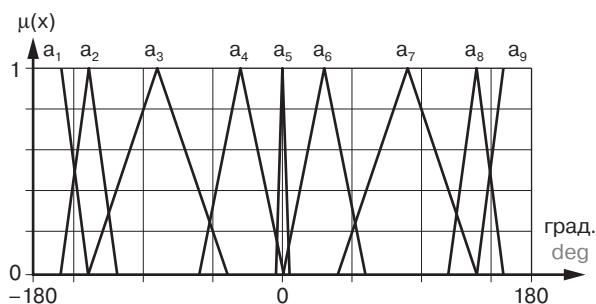


Рис. 2. Форма функции принадлежности входной переменной «ошибка угловой ориентации»

Fig. 2. The form of the membership function of the “angular orientation error” input variable

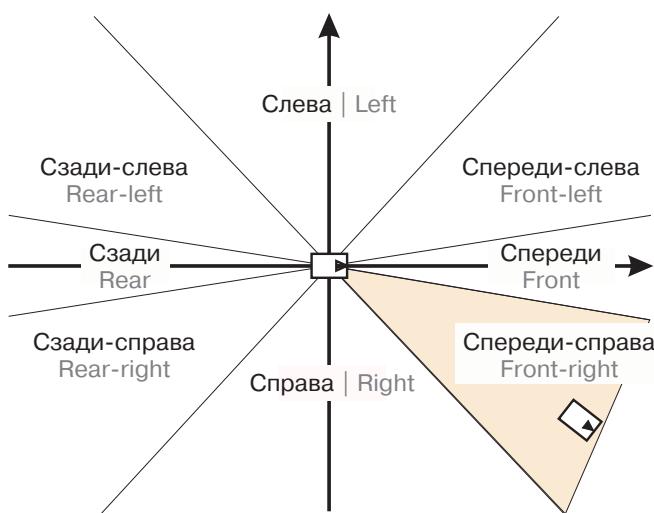


Рис. 3. Иллюстрация определения входной нечеткой переменной «сектор»

Fig. 3. An illustration of the definition of the second input fuzzy variable – “sector”

One of the problems of controlling mobile robots is the construction of the trajectory of motion. This task is one of the most important, since the construction of the optimal trajectory of motion makes it possible to increase the efficiency of the mobile robot operation. It should be noted that this problem can be solved using a control system based on the classical regulator [1]. However, the results do not always ensure a high performance of the robot.

At the same time, in conditions of autonomous control (lack of communication with the operator) the robot cannot receive corrective control from outside. Thus, the task must be solved from the beginning to the end directly by the robot control system. This solution should be made under certain limitations on the technical component (compactness, low power consumption), which affects the algorithmic support being used.

The object of control in this work is a robotic chassis. The chassis control system consists of electric motors and drives, as well as a computing device. Sensors are also a necessary part of the mobile robot. This paper features the work of the navigation system responsible for determining the orientation of the robot in space, taking into account the navigation system work's error.

Let us consider the solution of the problem of controlling the movement of a mobile robot from the point A with coordinates (x_n, y_n, α_n) to the point B with coordinates $(x_{n+1}, y_{n+1}, \alpha_{n+1})$, where x is the robot coordinate along the abscissa axis; y is the coordinate of the robot along the ordinate axis; α is the orientation angle of the robot (Fig. 1). The motion of a robot in a two-dimensional space without obstacles in real time is being modeled. The coordinates of the robot are determined in the Cartesian coordinate

именно положении должен оказаться робот в конце движения. Это позволит легко масштабировать систему управления для более сложных случаев движения среди препятствий, в узких проходах и коридорах.

Одним из вариантов решения этой проблемы является использование механизмов нечеткой логики. Эта концепция была предложена в 1965 году математиком Лотфи Заде и заключается в использовании нечетких множеств [2–4]. То есть таких множеств, характеристическая функция которых может принимать не только определенные значения — 0 или 1, но и любые другие в интервале [0, 1].

Предлагается использовать следующие переменные для реализации нечеткого алгоритма:

Входные переменные. Обозначим множество входных переменных $V = \{\beta_1, \beta_2\}$, где β_1 и β_2 переменные «ошибка угловой ориентации» и «сектор».

Переменная 1: β_1 — ошибка угловой ориентации робота. Определяется как разница между текущим углом ориентации робота и требуемым. В отличие от задачи следования траектории [5], эта переменная должна охватывать весь спектр значений от -180 до 180 . Таким образом, будут использованы следующие термы: a_1 — очень сильно отрицательная; a_2 — сильно отрицательная; a_3 — отрицательная; a_4 — слабо отрицательная; a_5 — близкая к нулю; a_6 — слабо положительная; a_7 — положительная; a_8 — сильно положительная; a_9 — очень сильно положительная. График функций принадлежности $\mu(x)$ этой переменной представлен на рис. 2.

Форма функций принадлежности не оказывает серьезного влияния на результат управления, что продемонстрировано в работе [6]. Таким образом, можно использовать распространенный вариант треугольных и трапецидальных функций принадлежности.

Переменная 2: β_2 — сектор. Представим систему координат, принятую таким образом, что положение мобильного робота является началом координат. Тогда направление, в котором находится точка назначения (примем следующие термы: b_1 — сзади-левее; b_2 — сзади-слева; b_3 — слева; b_4 — спереди-слева;

system, chosen for convenience reasons with further mapping of the trajectory. The peculiarity of the problem being solved is the consideration of the robot's orientation at the destination. The angle of the robot's orientation is the difference between the robot's motion vector and the abscissa axis in the chosen coordinate system. The trajectory of the movement will be constructed depending on the position of the robot at the end of the movement. This will make it easy to scale the control system for more complicated traffic cases among obstacles, narrow aisles and corridors.

One of the solutions to this problem is the use of fuzzy logic. This concept was proposed in 1965 by the mathematician Lotfi Zadeh and consists in using fuzzy sets [2 – 4]. That is, sets whose characteristic function can take not only certain values — 0 or 1, but also any others within the interval [0,1].

b_5 — спереди; b_6 — спереди-справа; b_7 — справа; b_8 — сзади-справа; b_9 — сзади-правее), и будет данной нечеткой переменной. Размеры секторов примем как 30 градусов для секторов «спереди» и «сзади», 60 градусов для секторов «слева» и «справа» и 75 градусов для остальных. Эта концепция проиллюстрирована на рис. 3.

Чтобы определить значение нечеткой переменной, нужно вычислить угол между направлением движения робота и направлением на точку назначения. Для этого преобразуем координаты обеих точек в систему отсчета, связанную с местонахождением робота (1) при помощи матрицы поворота (2):

$$\begin{pmatrix} x_1 \\ y_1 \\ \alpha_1 \end{pmatrix} = M(\alpha)^* \begin{pmatrix} x^* - x \\ y^* - y \\ \alpha^* - \alpha \end{pmatrix}, \quad (1)$$

где

$$M(\alpha) = \begin{pmatrix} \cos \alpha & \sin \alpha & 0 \\ -\sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{pmatrix}. \quad (2)$$

Тогда угол можно определить как (3):

$$S = \arctg \left(\frac{x_2 - x_1}{y_2 - y_1} \right). \quad (3)$$

График функций принадлежности этой переменной представлен на рис. 4.

Выходные переменные. Обозначим множество входных переменных $V = \{w_1, w_2\}$, где w_1 и w_2 переменные «желаемая угловая скорость» и «желаемая скорость».

Переменная 1: w_1 — желаемая угловая скорость (примем следующие термы: c_1 — быстро налево, c_2 — налево, c_3 — плавно налево, c_4 — прямо, c_5 — плавно направо, c_6 — направо, c_7 — быстро направо). График функций принадлежности этой переменной представлен на рис. 5.

Переменная 2: w_2 — желаемая линейная скорость (примем следующие термы: d_1 — полный назад, d_2 — назад, d_3 — малый назад,

It is suggested to use the following variables to implement a fuzzy algorithm:

Input variables. Let us denote the set of input variables $V = \{\beta_1, \beta_2\}$, where β_1 and β_2 are the variables “angular orientation error” and “sector”.

Variable 1: β_1 is error of the robot’s angular orientation. It is defined as the difference between the current orientation angle of the robot and the required one. In contrast to the task of trajectory [5], this variable should cover the whole range of values from -180 to 180 . Thus, the following terms will be used: a_1 — very strongly negative, a_2 — strongly negative, a_3 — negative, a_4 — slightly negative, a_5 — close to zero, a_6 — slightly positive, a_7 — positive, a_8 — strongly positive, a_9 — very strongly positive. Fig. 2 shows the graph of the membership functions of this variable.

The form of membership functions does not have a serious effect on the control result,

as demonstrated in [6]. In this way, you can use a common version of triangular and trapezoidal membership functions.

Variable 2: β_2 is sector. Let us visualize the coordinate system adopted in such a way that the position of the mobile robot is the origin of coordinates. Then the direction in which the destination is located (take the following terms: b_1 — rear-left, b_2 — left-rear, b_3 — left, b_4 — front-left, b_5 — front, b_6 — front-right, b_7 — right, b_8 — right-rear, b_9 — rear-right), will be given a fuzzy variable. Sizes of sectors will be taken as 30 degrees for the sectors “front” and “rear”, 60 degrees for the sectors “left” and “right” and 75 degrees for the others. This concept is illustrated in Fig. 3.

To determine the value of a fuzzy variable, we need to calculate the angle between the direction of the robot’s movement and the direction towards the destination point. To do this, we transform the coordinates of both points into the reference frame

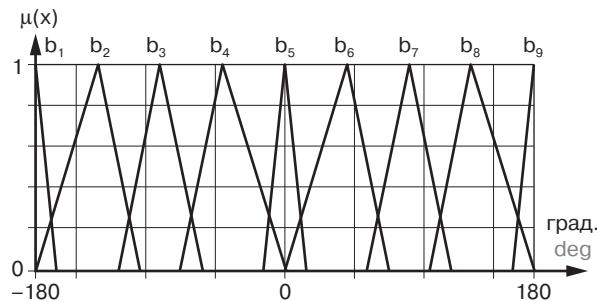


Рис. 4. Форма функции принадлежности входной переменной «сектор»

Fig. 4. The form of the membership function of the “sector” input variable

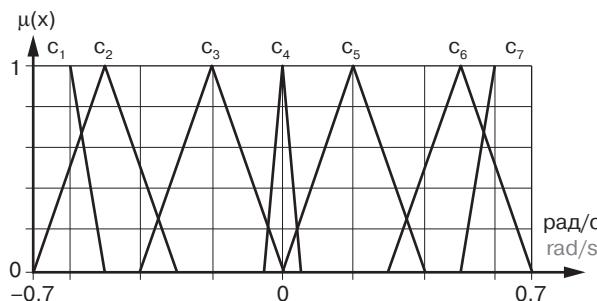


Рис. 5. Форма функции принадлежности выходной переменной «желаемая угловая скорость»

Fig. 5. The form of the membership function of the “desired angular velocity” output variable

d_4 — стоп, d_5 — малый вперед, d_6 — вперед, d_7 — полный вперед). График функций принадлежности этой переменной представлен на рис. 6.

Формирование базы правил нечеткой системы заключается в формировании ряда высказываний вида [7]:

ЕСЛИ « β_1 есть a_i » И « β_2 есть b_j » ТО « w_1 есть c_p » И « w_2 есть d_q »,

connected with the location of the robot (1) using the rotation matrix (2):

$$\begin{pmatrix} x_1 \\ y_1 \\ \alpha_1 \end{pmatrix} = M(\alpha)^* \begin{pmatrix} x^* - x \\ y^* - y \\ \alpha^* - \alpha \end{pmatrix}, \quad (1)$$

where

$$M(\alpha) = \begin{pmatrix} \cos \alpha & \sin \alpha & 0 \\ -\sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{pmatrix}. \quad (2)$$

Then the angle can be defined as (3):

$$S = \arctg \left(\frac{x_2 - x_1}{y_2 - y_1} \right). \quad (3)$$

The graph of the membership functions of this variable is shown in Fig. 4.

Output variables. Let us denote the set of input variables $V = \{w_1, w_2\}$, where w_1 and w_2 are the variables “desired angular velocity” and “desired speed”.



где a и b — термы входных переменных, c и d — термы выходных переменных, i, j, p и q — индексы соответствующих термов.

База правил должна охватывать взаимные отношения всех термов всех входных переменных и описывать реакцию системы на те или иные условия.

Для решаемой задачи база правил составила 81 правило, некоторые из которых приведены ниже:

...

II ЕСЛИ «Ошибка угловой ориентации» есть «сильно отрицательная» И «Сектор» есть «сзади» ТО «Желаемая угловая скорость» есть «быстро направо» И «Желаемая линейная скорость» есть «назад».

...

48 ЕСЛИ «Ошибка угловой ориентации» есть «близкая к нулю» И «Сектор» есть «спереди-справа» ТО «Желаемая угловая скорость» есть «направо» И «Желаемая линейная скорость» есть «вперед».

...

71 ЕСЛИ «Ошибка угловой ориентации» есть «положительная» И «Сектор» есть «сзади-справа» ТО «Желаемая угловая скорость» есть «быстро направо» И «Желаемая линейная скорость» есть «вперед».

...

Существует несколько алгоритмов нечеткого вывода, каждый из которых имеет свои особенности. Основных алгоритмов четыре:

- алгоритм Мамдани [8];
- алгоритм Такаги – Сугено;
- алгоритм Цукамото;
- алгоритм Ларсена.

Ряд ограничений, свойственных для каждого из алгоритмов, позволяет определить наиболее подходящий для решаемой задачи [9].

В частности, алгоритм Цукамото используется только для монотонных функций принадлежности выходных переменных. Это автоматически делает его непригодным в описываемых

Variable 1: w_1 is the desired angular velocity (take the following terms: c_1 — fast left, c_2 — left, c_3 — smoothly left, c_4 — straight, c_5 — fast right, c_6 — right, c_7 — smoothly right). Fig. 5 shows the graph of the membership functions of this variable.

Variable 2: w_2 is desired speed (take the following terms: d_1 — full back, d_2 — back, d_3 — small back, d_4 — stop, d_5 — small forward, d_6 — forward, d_7 — full forward). Fig. 6 shows the graph of the membership functions of this variable.

Formation of the fuzzy system rules base is the formation of a number of utterances of the form [7]:

IF “ β_i is a_i ” AND “ β_j is b_j ” THEN “ w_1 is c_p ” AND “ w_2 is d_q ”,

where a and b are the terms of the input variables, c and d are the terms of the output variables, i, j, p and q are the indices of the corresponding terms.

The base of rules should cover the mutual relations of all the terms of all input variables

and describe the reaction of the system to certain conditions.

For the problem to be solved, the rule base consisted of 81 rules, some of which are given below:

...

II IF “angular orientation error” is “strongly negative” AND “sector” is “rear” THEN “desired angular velocity” is “fast right” AND “desired speed” is “back”

...

48 IF “angular orientation error” is “close to zero” AND “sector” is “front-right” THEN “desired angular velocity” is “right” AND “desired speed” is “forward”

...

71 IF “angular orientation error” is “positive” AND “sector” is “rear-right” THEN “desired angular velocity” is “fast right” AND “desired linear speed” is “forward”

...

There are several fuzzy algorithms, each of which has its own peculiarities. The main algorithms are four:

- Mamdani algorithm [8];
- Takagi-Sugeno algorithm;
- Tsukamoto algorithm;
- Larsen's algorithm.

A number of limitations inherent in each of the algorithms make it possible to determine the most suitable for the problem being solved [9].

In particular, the Tsukamoto algorithm is used only for monotonic membership functions of the output parameter. This automatically makes it unsuitable under the conditions described, when one of the variables is the angular velocity, whose value varies from negative to positive values [10].

The Takagi – Sugeno algorithm is used when the weight coefficients are known for the unknown form of the output variables of membership functions. Since in our case

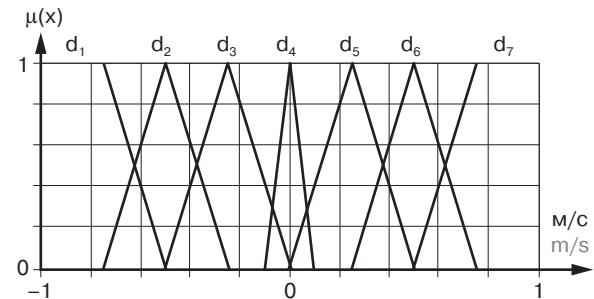


Рис. 6. Форма функции принадлежности выходной переменной «желаемая линейная скорость»

Fig. 6. The form of the membership function of the output variable “desired linear velocity”

условиях, когда одной из переменных является угловая скорость, значение которой колеблется от отрицательных до положительных значений [10].

Алгоритм Такаги – Сугено применяется, когда известны весовые коэффициенты при неизвестной форме функций принадлежности выходных переменных. Так как в нашем случае мы можем выбрать форму функций, нет необходимости использовать нечеткий вывод Сугено [11].

Алгоритмы Мамдани и Ларсена похожи по своей структуре и отличаются лишь способом выбора импликации. Рассмотрим алгоритм Мамдани подробнее (рис. 7).

Суть фазификации заключается в приведении в соответствие четких значений, подающихся на вход нечеткой системы и термов нужной входной переменной. Таким образом, вводится ряд высказываний вида « β_i есть a_i », « β_j есть a_j » и т. д. Для каждого из них устанавливается истинность a_n^* путем подстановки значений a_n в функцию принадлежности $\mu(x)$. После того как все значения $a_n^* = \mu(a_n)$ для всех термов всех входных переменных найдены, этап фазификации считается законченным.

На этапе агрегирования определяется степень истинности для каждого правила из базы. Так как правила в базе состоят из простых высказываний « β_i есть a_n », то при этом используются результаты фазификации.

Активизация представляет собой процесс определения степени истинности для каждого из элементарных логических подзаключений, которые вычисляются на основе выходных переменных. Существуют различные методы активизации:

- min-активизация — $\mu'(c) = \min\{f_i, \mu(c)\}$
- prod-активизация — $\mu'(c) = f_i \cdot \mu(c)$
- average-активизация — $\mu'(c) = 0.5(f_i + \mu(c))$,

где множество $F = (f_1, f_2, \dots, f_q)$ — множество степеней истинности для каждого из правил; q — общее количество подзаключений в базе правил; $\mu(c)$ — функция принадлежности терма, который является значением некоторой выходной переменной.

В алгоритме Мамдани используется min-активизация, в алгоритме Ларсена — prod-активизация. В ряде случаев [12] алгоритм Ларсена оказывается точнее алгоритма Мамдани, но требует больше операций умножения.

Аккумуляция — это нахождение функции принадлежности для каждой из выходных переменных путем объединения всех степеней истинности подзаключений, полученных ранее.

Финальная часть любого нечеткого алгоритма — переход от функции принадлежности выходных переменных к их численным (четким) значениям. Таким образом, результаты нечеткого алгоритма можно будет использовать в последующих вычислениях.

Моделирование движения мобильного робота и нечеткой системы управления проводилось в среде графического программирования LabView. LabView по умолчанию поддерживает работу с нечеткими системами, что лишает необходимости непосредственно реализовывать нечеткие алгоритмы с нуля, при этом оставляя все возможности для модернизации этих алгоритмов.

Блок-диаграмма модели представлена на рис. 8. Помимо непосредственно нечеткой системы (рис. 9), здесь присутствуют блоки определения сектора, ошибки ориентации, ошибки положения, преобразования координат, эмуляции ошибки навигации.

Рассмотрим результаты моделирования перемещения робота от точки к точке. Функционирование нечеткого регулятора сравнивались с функционированием модели из работы [1].

we can choose the form of the functions, there is no need to use the fuzzy derivation of Sugeno [11].

The algorithms of Mamdani and Larsen are similar in their structure and differ only in the way they choose the implication. Let us consider the Mamdani algorithm in detail (Fig. 7).

The essence of fuzzification is to bring into correspondence the precise values that are fed to the input of the fuzzy system and the terms of the required input variable. Thus, a number of statements of the form " β_i is a_1 ", " β_i is a_2 " etc. are entered. For each of them, the truth of a_n^* is established by substituting the values of a_n in the membership function $\mu(x)$. After all the values of $a_n^* = \mu(a_n)$ for all terms of all input

variables are found, the phase of fuzzification is considered complete.

At the stage of aggregation, the degree of truth for each rule from the database is determined. Since the rules in the database consist of simple statements " β_i is a_n ", the results of fuzzification are used.

Activation is the process of determining the degree of truth for each of the elementary logical sub-clauses, which are calculated on the basis of the output variables. There are various methods of activation:

- min-activation — $\mu'(c) = \min\{f_i, \mu(c)\}$
- prod-activation — $\mu'(c) = f_i \cdot \mu(c)$
- average-activation — $\mu'(c) = 0.5(f_i + \mu(c))$, where $F = (f_1, f_2, \dots, f_q)$ is the set of truth degrees for each of the rules; q is the total

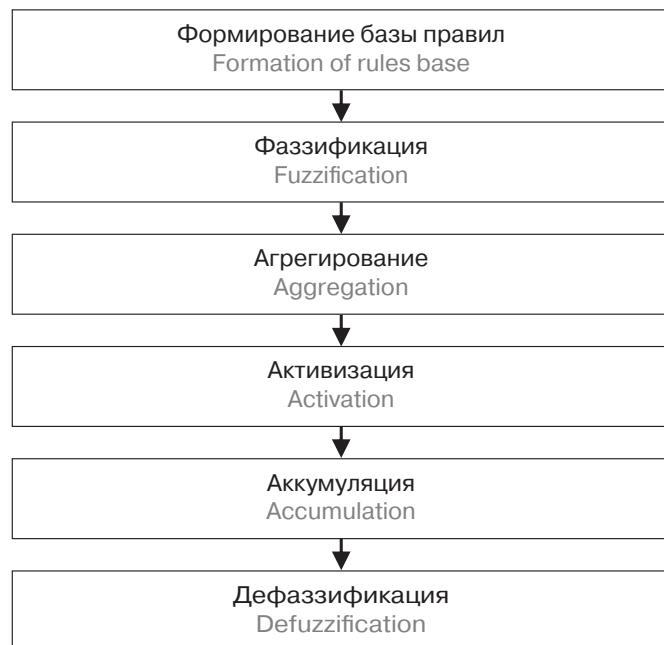


Рис. 7. Последовательность шагов нечеткого вывода Мамдани
Fig. 7. The sequence of steps of Mamdani fuzzy algorithm

Моделирование показывает (рис. 10), что робот достигает заданной точки. Можно увидеть, что во второй модели робот движется по криволинейной траектории, что усложняет управление и стабилизацию, а также приводит к более быстрому износу исполнительных механизмов. Нечеткая модель придерживается движения по прямым отрезкам на большей части пути, что является ее преимуществом.

На рис. 11 показано движение робота по прямой линии из точки с координатами $(0; 0)$ в точку с координатами $(200; 0)$. В данном эксперименте также было проведено сравнение с моделью из работы [1]. В обоих случаях робот достигает цели, но нечеткая система показывает меньшее отклонение ($0,5$ метров против $1,0$ метра у классического регулятора) от прямой линии.

Рис. 12 показывает результаты моделирования разворота робота из точки $(0; 0)$ с ориентацией 180 градусов в точку $(10; 0)$.

number of sub-clauses in the rules database; $\mu(c)$ is the membership function of the term, which is the value of some output variable.

In the Mamdani algorithm, min-activation is used, in the Larsen algorithm — prod-activation. The paper [12] shows that in a number of cases the Larsen algorithm is more accurate than the Mamdani algorithm, but it requires more multiplication operations.

Accumulation is finding the membership function for each of the output variables by combining all the degrees of truth of the sub-clauses obtained earlier.

The final part of any fuzzy algorithm is the transition from the function of the output variables to their numerical (precise) values. Thus, the results of the fuzzy algorithm

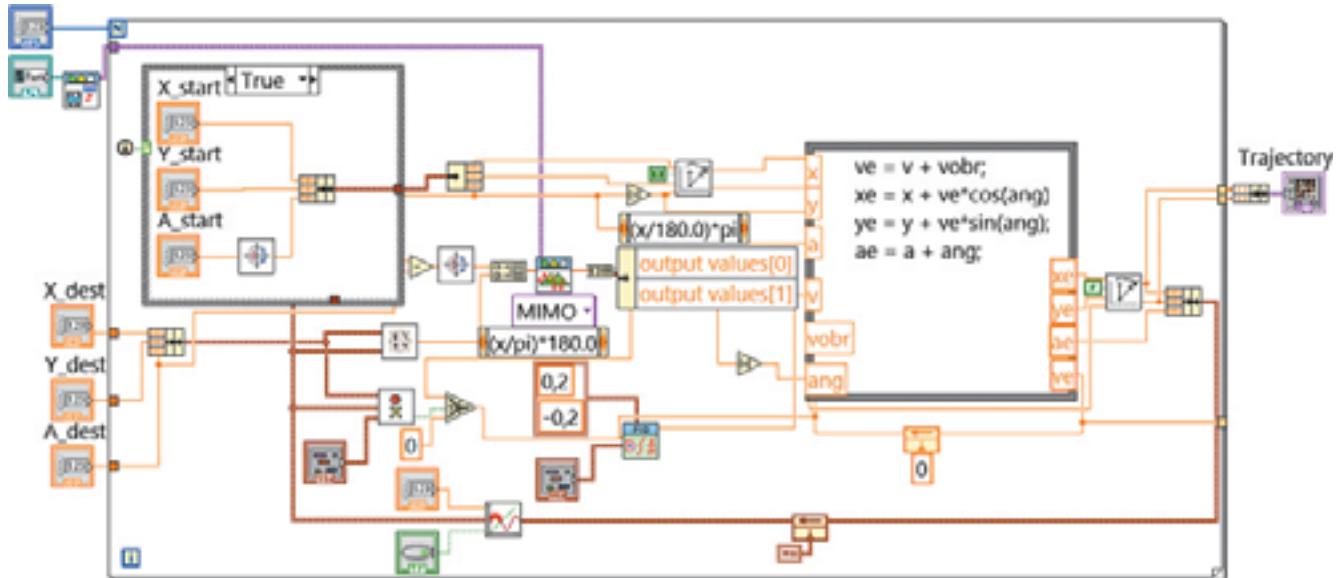


Рис. 8. Блок-диаграмма модели движения мобильного робота, построенной на основе нечеткой логики
Fig. 8. Block diagram of the mobile robot motion model built on the basis of fuzzy logic

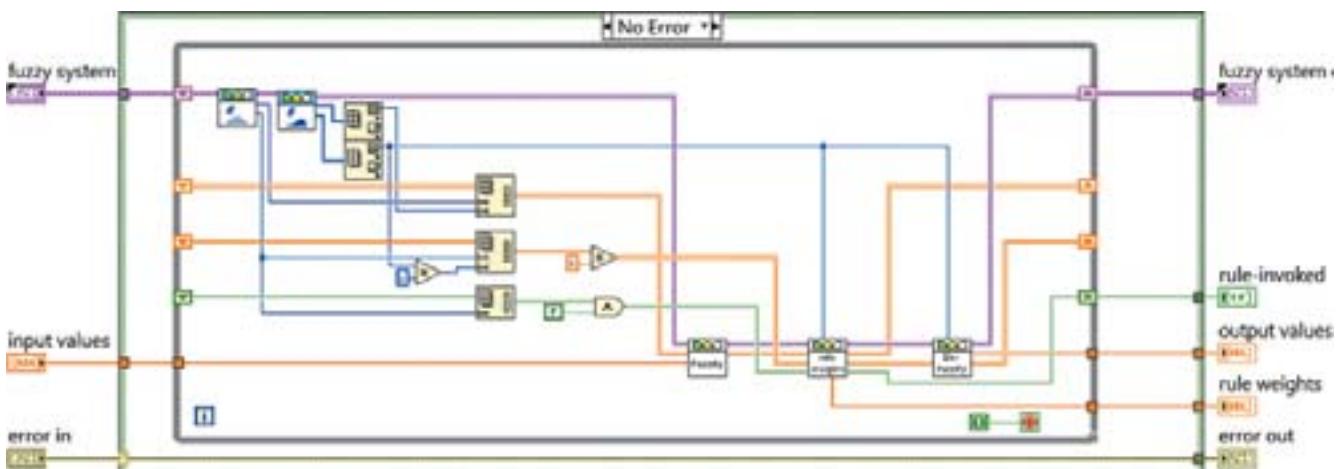


Рис. 9. Блок-диаграмма нечеткой системы управления
Fig. 9. Block diagram of the fuzzy control system

can be used in subsequent precise calculations.

Simulation of the movement of the mobile robot and fuzzy control system was carried out in the LabView programming environment. LabView supports the work with fuzzy systems by default, which makes it unnecessary to directly implement fuzzy algorithms from scratch, while leaving all the possibilities for upgrading these algorithms.

The block diagram of the model is shown in Fig. 8. In addition to the directly fuzzy system (Fig. 9), there are sector determination blocks, orientation errors, position errors, coordinate transformations, and navigation error emulation.

Let's consider the results of simulating the robot's moving from point to point.

The operation of the fuzzy controller was compared with the model from [1]. The simulation shows (Fig. 10) that a robot controlled by both models reaches the specified point. It can be seen that the second model moves along a curvilinear trajectory, which complicates control and stabilization, and also results in the faster wear of the actuators. Most of the way the fuzzy model adheres to the movement along straight lines, which is its advantage.

Fig. 11 shows the movement of the robot along a straight line from point with coordinates (0; 0) to point with coordinates (200; 0). In this experiment, a comparison was also made with the model from [1]. In both cases, the robot reaches the target, but the fuzzy

system shows a smaller deviation (0.5 meters vs. 1.0 meter for the classic regulator) from the straight line.

Fig. 12 shows the results of simulating a robot turn from point (0; 0) with an orientation of 180 degrees to point (10; 0) with an orientation of 0 degrees. The peculiarity of this maneuver is that the distances to be covered do not exceed 10 meters in both directions. Thus, the error of the navigation system (NS) plays an important role. The global NS of the robot has an error of up to 5 meters. In the case of moving to much larger distances, this, as can be seen from previous experiments, does not play a major role. But in the case of a turn around, the orientation of the robot with such an error

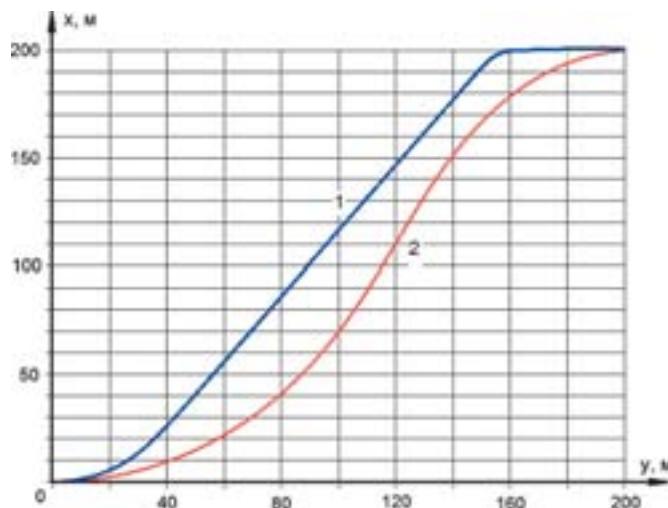


Рис. 10. Моделирование перемещения робота из точки $(0; 0)$ с ориентацией 0 градусов в точку $(200; 200)$ с ориентацией 0 градусов: 1 — модель на основе нечеткой логики; 2 — модель из работы [4]
Fig. 10. Modelling the movement of a robot from point $(0; 0)$ with an orientation of 0 degrees to point $(200; 200)$ with an orientation of 0 degrees: 1 — model based on fuzzy logic; 2 is a model from [4]

с ориентацией 0 градусов. Особенностью данного маневра является то, что преодолеваемые расстояния не превышают 10 метров в обоих направлениях. Таким образом, большую роль имеет погрешность навигационной системы (НС). Глобальная НС робота имеет ошибку до 5 метров. Как видно из предыдущих экспериментов, при движении на большие расстояния эта ошибка не играет существенной роли. Но в случае разворота ориентирование робота с такой ошибкой становится невозможным, поскольку реальное положение робота и положение, определяемое НС, могут отличаться настолько, что будут находиться по разные стороны от точки назначения. Это приводит к тому, что робот не может достичь цели и беспорядочно передвигается вокруг точки старта. Траектория движения робота при попытке развернуться в таких условиях изображена на графике 1 рис. 12.

Поэтому мобильный робот должен иметь также автономные средства ориентирования (например, инерциальные навигационные системы, лазерные дальномеры) с погрешностью порядка десятков сантиметров. Использование таких НС при маневрах рядом с точкой назначения позволит более точно ориентировать робота и достичь поставленной цели, что проиллюстрировано на графике 2 рис. 12.

В результате работы была построена модель системы управления (регулятор) мобильным роботом на основе нечеткой логики, выбраны входные нечеткие переменные «ошибка угловой ориентации» и «сектор». Проведено компьютерное моделирование в среде графического программирования LabView. Представлены результаты сравнения работы моделей на основе нечеткого регулятора и классического регулятора, а также различные траектории движения робота.

Авторы считают, что в данной работе новой является модель нечеткого регулятора на основе нечетких переменных «ошибка угловой ориентации» и «сектор», с помощью которой получены результаты сравнения функционирования классического регулятора и регулятора, построенного с применением нечеткой логики. Разработка модели и проведение компьютерных экспериментов

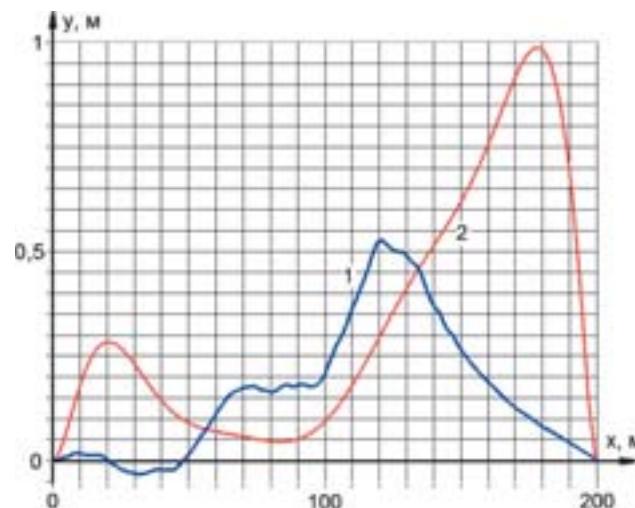


Рис. 11. Моделирование движения робота по прямой из точки $(0; 0)$ с ориентацией 0 градусов в точку $(200; 0)$ с ориентацией 0 градусов: 1 — модель на основе нечеткой логики; 2 — модель из работы [1]
Fig. 11. Simulation of the robot's motion along a straight line from the point $(0; 0)$ with the orientation of 0 degrees to the point $(200; 0)$ with the orientation of 0 degrees: 1 — model based on fuzzy logic; 2 is a model from [1]

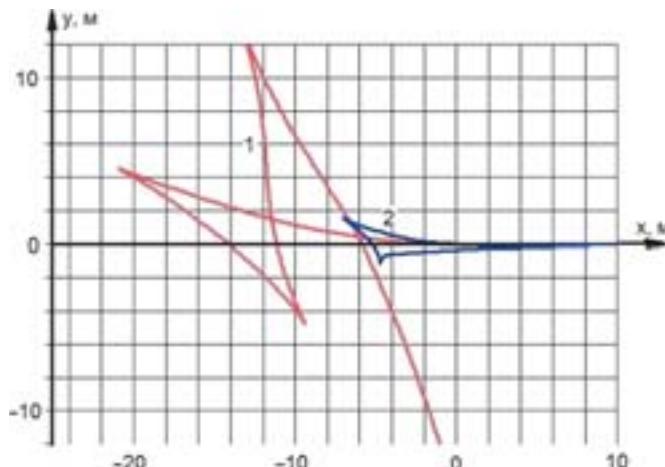


Рис. 12. Моделирование разворота робота: 1 — с погрешностью навигационной системы 5 метров; 2 — с погрешностью навигационной системы 0,2 метра
Fig. 12. Modeling the turn of the robot: 1 — with a navigation system error of 5 meters; 2 — with a navigation system error of 0.2 meters

проводились в среде графического программирования LabView. Полученные результаты подтвердили адекватность используемой модели и разработанного набора правил работы нечеткого регулятора.

Работа выполнена при финансовой поддержке Минобрнауки России, проект № 8.8952.2017/8.9.

ЛИТЕРАТУРА

1. Vinogradov A., Terentev A., Petrov V. and Petrov O. «Development of Mathematical Model of Moving Wheeled Robot Using Visual Programming Platform Labview», 2017 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EICon-Rus), St. Petersburg, 2017, pp. 1056–1059.



2. Финаев В. И., Игнатьев В. В. Системы управления на основе объединения классической и нечеткой моделей объекта. — М.: Физматлит, 2013. — 158 с.
3. Ходашинский И. А. Идентификация нечетких систем: методы и алгоритмы // Проблемы управления, 2009, № 4.
4. Цой Дуньюэ. Разработка метода планирования траектории перемещения мобильного автономного робота в трехмерной среде на основе аппарата нечеткой логики // Инженерный вестник Дона, № 4, 2015.
5. Цой Дуньюэ. Управление мобильным роботом на основе нечетких моделей // Российская Академия Естествознания, 2007, № 6.
6. Чернов В. Г. Модификация алгоритмов управления, использующих правила нечеткого условного вывода // Информационно-управляющие системы, № 3, 2013. — С. 23–29.
7. Семенова Н. Г., Семенов А. М., Крылов И. Б. База знаний интеллектуальной обучающей системы технической дисциплины // Вестник Оренбургского государственного университета, 2013, № 9. — С. 232–239.
8. Семченко П. Н. Основанные на правилах экспертные системы // Электронное научное издание «Ученые заметки ТОГУ» 2014, Том 5, № 4. — С. 1249–1266.
9. Леденева Т. М., Кашко В. В. О различных реализациях механизма нечеткого логического вывода // Актуальные проблемы прикладной информатики, информатики и механики: сборник трудов Международной научно-технической конференции, Воронеж, 2016. — С. 391.
10. Ткач С. Н. Настройка параметров адаптивного контроллера с использованием нечеткой нейронной сети // Программные продукты и системы. 2001. № 3.
11. Семенова Н. Г., Крылов И. Б. Разработка агентно-ориентированной интеллектуальной обучающей системы на основе нечеткой нейронной сети Такаги – Сугено – Канга // Вектор науки ТГУ, 2015, № 2-1 (32-1).
12. Мелков Д. А. Сравнение алгоритмов нечеткого вывода с использованием языков стандарта МЭК // Молодой ученый, 2013, № 5. — С. 74–79.

becomes impossible, since the actual position of the robot and the position determined by the NS can be so different that they will be on different sides of the destination. This leads to the fact that the robot can't reach the target and randomly moves around the starting point. The trajectory of the robot's movement when trying to turn around under such conditions is shown in the graph 1 of Fig. 12.

Therefore, the mobile robot should also have autonomous means of orientation (for example, inertial navigation systems, laser rangefinders) with an error of the order of tens of centimeters. The use of such NS at maneuvers near the destination will allow more precise orientation of the robot and achieve the goal, which is illustrated in graph 2 of Fig. 12.

As a result of the work, a model of a regulator controlling a mobile robot based on fuzzy logic has been constructed, input fuzzy "angular orientation error" and "sector" variables have been chosen. Computer simulation in LabView's graphical programming environment has been carried out. The results of comparison of the work of models on the basis of a fuzzy controller and a classical regulator have been presented, as well as various trajectories of the robot's movement.

The authors consider novel the model of a fuzzy regulator based on fuzzy variables "angular orientation error" and "sector". With the help of this model, we have obtained the results of comparison of functioning of a classical regulator and that of a regulator constructed using fuzzy logic. The development of the model and the

conduct of computer experiments were carried out in the LabView graphical programming environment. The obtained results have confirmed the adequacy of the model used and the developed set of rules for the operation of the fuzzy controller.

The work was carried out with the financial support of the Ministry of Education and Science of Russia, project No. № 8.8952.2017/8.9.

REFERENCES

1. Vinogradov A., Terentev A., Petrov V. and Petrov O. "Development of Mathematical Model of Moving Wheeled Robot Using Visual Programming Platform Labview", 2017 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EICONRUS), St. Petersburg, 2017, pp. 1056–1059.
2. Finaev V.I., Ignat'ev V.V. *Sistemy upravleniya na osnove ob"edineniya klassicheskoi i nechetkoi modelei ob"ekta*. M.: Fizmatlit, 2013. 158 p. (In Russian).
3. Khodashinskii I.A. *Identifikatsiya nechetkikh sistem: metody i algoritmy* // Problemy upravleniya, 2009, № 4. (In Russian).
4. Tsuji Dun'ye. *Razrabotka metoda planirovaniya traektorii peremeshcheniya mobil'nogo avtonomnogo robota v trekhmernoi srede na osnove apparata nechetkoi logiki* // Inzhenernyi vestnik Doma, № 4, 2015. (In Russian).
5. Tsuji Dun'ye. *Upravlenie mobil'nym robotom na osnove nechetkikh modelei* // Rossiiskaya Akademiya Estestvoznaniya, 2007, № 6. (In Russian).
6. Chernov V.G. *Modifikatsiya algoritmov upravleniya, ispol'zuyushchikh pravila nechetkogo uslovnogo vypovedi* // *Informatsionno-upravlyayushchie sistemy*, № 3, 2013. P. 23–29. (In Russian).
7. Semenova N. G., Semenov A. M., Krylov I. B. *Baza znanii intellektual'noi obuchayushchei sistemy tekhnicheskoi discipliny* // Vestnik Orenburgskogo gosudarstvennogo universiteta, 2013, № 9. P. 232–239. (In Russian).
8. Semchenko P. N. *Osnovannye na pravilakh ekspertnye sistemy* // Elektronnoe nauchnoe izdanie "Uchenye zametki TOGU" 2014, Tom 5, № 4. P. 1249–1266. (In Russian).
9. Ledeneva T. M., Kashko V. V. *O razlichnykh realizatsiyakh mehanizma nechetkogo logicheskogo vypovedi* // Aktual'nye problemy prikladnoi informatiki, informatiki i mehaniki: sbornik trudov Mezhdunarodnoi nauchno-tehnicheskoi konferentsii, Voronezh, 2016. P. 391. (In Russian).
10. Tkach S. N. *Nastroika parametrov adaptivnogo kontrollera s ispol'zovaniem nechetkoi neironnoi seti* // Programmnye produkty i sistemy. 2001. № 3. (In Russian).
11. Semenova N. G., Krylov I. B. *Razrabotka agentno-orientirovannoj intellektual'noj obuchayushchei sistemy na osnove nechetkoi neironnoi seti Takagi – Sugeno – Kanga* // Vektor nauki TGU, 2015, № 2-1 (32-1). (In Russian).
12. Melkov D. A. *Sravnenie algoritmov nechetkogo vypovedi s ispol'zovaniem yazykov standarta MEK* // Molodoi uchenyi, 2013, № 5. P. 74–79. (In Russian).