

## УПРАВЛЕНИЕ МОБИЛЬНЫМ РОБОТОМ «ARCBOTICS SPARKI» МЕТОДОМ КОНЕЧНОГО СОСТОЯНИЯ

Е.А. Шушляпин, А.А. Афолина, А.Е. Безуглая

ФГАОУ ВО «Севастопольский государственный университет»,  
РФ, г. Севастополь, ул. Университетская, 33  
E-mail: bu6@bk.ru

В работе выполнены исследования по разработке и верификации алгоритмов управления мобильным двухколесным роботом ArcBotics Sparki с целью его приведения в точку с заданными координатами и произвольным углом подхода. Предложен критерий, учитывающий нелинейную кинематическую связь между углом подхода и декартовыми координатами текущего положения робота. Разработаны алгоритмы управления на основе непрерывного и дискретного вариантов метода конечного состояния.

**Ключевые слова:** мобильные двухколесные двухколейные роботы, нелинейные системы, терминальное управление, линеаризационный по входу/выходу метод, метод конечного состояния, математическое программирование.

**Введение.** Двухколесные двухколейные роботы применяются в качестве амфибийных платформ, используемых для наблюдения и измерения. Например, в описании патента [1] подобный робот, содержащий «...два колесных движителя, расположенных на бортах платформы и закрепленных на осях двигателей, расположенных внутри полой платформы», предлагается использовать, в частности, для неразрушающего контроля, измерений посредством лазера, а также телевизионного наблюдения, при периодическом осмотре поверхностей подводной части гидротехнической и нефтегазопромышленной инфраструктуры.

В работе [2] приведена математическая модель двухколесного двухколейного робота и построен алгоритм терминального управления роботом на основе использования метода конечного состояния [3]. Ввиду специфики математической модели робота получен алгоритм, представляющий собой совокупность алгебраических соотношений. Реализация управления на реальном оборудовании – роботе ArcBotics Sparki, – показала работоспособность алгоритма для случаев, когда дискретный такт при расчете управления не меньше  $0,1с$ . Такой такт, в сочетании с относительной сложностью алгоритма и необходимостью расчета текущего положения робота методом счисления с использованием его математической модели, побуждает ис-

пользовать управления, рассчитываемые в дискретные моменты времени [4]. При этом расчет текущего состояния можно производить с малым шагом, а расчет управления – с большим шагом. Целью работы является применение метода конечного состояния [3] в непрерывной и дискретной формах для приведения робота в заданную точку.

**Математические модели и методы управления.** Кинематическая модель робота имеет вид [2]

$$\begin{aligned} \frac{dx_1}{dt} &= \frac{D_w(\omega_1 + \omega_2)}{4} [\cos \varphi], \\ \frac{dx_2}{dt} &= \frac{D_w(\omega_1 + \omega_2)}{4} [\sin \varphi], \\ \frac{d\varphi}{dt} &= \frac{D_w(\omega_1 - \omega_2)}{2L_r}, \end{aligned} \quad (1)$$

где  $x_1, x_2$  – линейные перемещения робота вдоль координатных осей;  $D_w$  – диаметр первого и второго колеса;  $\omega_1, \omega_2$  – угловые скорости вращения первого и второго колеса (управляющие воздействия);  $L_r$  – расстояние между колесами.

**Непрерывное управление.** В [2] для приведения робота в заданную точку за заданное время использованы линеаризационный по входу/выходу метод А. Isidori в модификации Wu Wei и метод конечного состояния [3], которые, ввиду

специфики модели (1), привели к одному и тому же алгоритму

$$\begin{aligned} u(t) &= G^+ \left( \frac{J^* - J(x(t))}{T} \right), \\ G^+ &= G^T (GG^T)^{-1}, \\ G &= \frac{\partial J(x(t))}{\partial x} B(t, x(t)), \end{aligned} \quad (2)$$

где  $J(t) \equiv J(x(t))$  – критериальная функция, определяющая заданную экспоненциальную с постоянной времени  $T_u$  траекторию, дифференциальное уравнение для которой следующее

$$\frac{dJ(t)}{dt} = \frac{J^* - J(t)}{T_u}. \quad (3)$$

В уравнениях (2) переменные состояния  $x = \{x_i\}, i = 1, \dots, n$  и управления  $u = \{u_j\}, j = 1, \dots, r$  модели (1) переобозначены в соответствии

$$x = \begin{pmatrix} x_1 \\ x_2 \\ \varphi \end{pmatrix}, \quad u = \begin{pmatrix} \omega_1 \\ \omega_2 \end{pmatrix}. \quad (4)$$

Для критериальной функции использовалась квадратичная форма вида

$$\begin{aligned} J(x(t)) &= F_1 (x_1(t) - x_1^*)^2 + \\ &+ F_2 (x_2(t) - x_2^*)^2 + \\ &+ F_3 (x_3(t) - x_3^*)^2 \rightarrow J^* = 0, \end{aligned} \quad (5)$$

смысл которой в том, что при приближении текущего состояния  $x(t)$  к заданной точке  $x^*$  значение критериальной функции стремится к заданному значению  $J^*$ , которое, очевидно, должно быть равно нулю. Эта цель при управлении (2) может оказаться недостижимой, что приведет к отклонению фактического поведения (5) от желаемой траектории (3). Основная причина недостижимости – равенство нулю скаляра  $GG^T$  в отдельных точках временной оси либо на интервалах. Это, как бы, естественная при-

чина недостижимости, вызванная несоответствием цели и способом ее достижения по траектории (3). Другая возможная причина отклонения фактической траектории от заданной – амплитудные ограничения на управляющие воздействия, которые в использованных методах не могут быть учтены. Так, скорости вращения колес робота ограничены по амплитуде величиной  $\omega_m$

$$\omega_i = \begin{cases} -\omega_m, & u_i < -\omega_m, \\ u_i, & -\omega_m \leq u_i \leq \omega_m, \\ \omega_m, & u_i > \omega_m, \end{cases} \quad (6)$$

$$i = 1, 2.$$

Выход рассчитываемых по (2) управлений на ограничения (6) также искажает форму критериальной функции, определяемой уравнением (3), но при этом все равно критериальная функция уменьшается, стремясь к какому-то минимально достижимому значению  $J^*$ . При этом достигнутое значение  $J^*$  определяет точность приведения компонент вектора состояния к заданным значениям.

Система (1) представляет собой нелинейную систему аффинного типа с двумя управляющими воздействиями  $\omega_1, \omega_2$ , векторно-матричная запись которой имеет вид

$$\frac{dx}{dt} = Ax + B(x)u, \quad (7)$$

где  $A$  – нулевая матрица, а компоненты матрицы  $B = \{B_{ij}\}, i = 1, 2, 3, j = 1, 2$  следующие

$$\begin{aligned} B_{11} &= B_{12} = \frac{D_w \cos(\varphi)}{4}, \\ B_{21} &= B_{22} = \frac{D_w \sin(\varphi)}{4}, \\ B_{31} &= -B_{32} = \frac{D_w}{2L_r}. \end{aligned} \quad (8)$$

В [2] приведены результаты Mathcad-моделирования (1) с управлением (2), критерием (5) и амплитудными ограни-

чениями (6) применительно к роботу ArcBotics Sparki с параметрами:  $D_w=0,05$ ;  $L_r=0,085$ ;  $F_1=10$ ;  $F_2=0$ ;  $F_3=1$ ;  $x_1^*=0,6$  м;  $x_2^*=0,3$  м;  $x_3^*=0,8$  м;  $\omega_w=1,5$  с<sup>-1</sup>;  $h=0,1$  с;  $T_u=0,01$  с;  $t_f=40$  с.

Этот же алгоритм реализован в бортовой версии языка C++ реального робота. Как следует из кинематической схемы, на основе которой составлены уравнения (1), между состояниями модели (1) имеется нелинейная зависимость

$$\operatorname{tg} \varphi = \frac{x_2}{x_1}. \quad (9)$$

Следовательно, уравнения (1) являются нелинейно зависимыми, что означает следующее: при реализации управления (2) можно одновременно достичь заданных значений либо парой  $x_1, \varphi$ , либо парой  $x_2, \varphi$ , а вот для пары  $x_1, x_2$  подобная цель в общем случае недостижима.

Это хорошо видно, если преобразовать (1), приняв в качестве новых управляющих воздействий

$$v_1 = \frac{\omega_1 + \omega_2}{2}, \quad v_2 = \frac{\omega_1 - \omega_2}{2}.$$

В результате получим систему

$$\begin{aligned} \frac{dx_1}{dt} &= \frac{D_w}{2} v_1 [\cos \varphi], \\ \frac{dx_2}{dt} &= \frac{D_w}{2} v_1 [\sin \varphi], \\ \frac{d\varphi}{dt} &= \frac{D_w}{L_r} v_2. \end{aligned} \quad (10)$$

Из уравнений (10) видно, что управление  $v_2$  воздействует только на  $\varphi$ , а управление  $v_1$  — одновременно на  $x_1$  и  $x_2$ . При этом, если при выбранном  $v_2$  достигается некоторое значение  $\varphi$ , то далее можно выбирать  $v_1$  при известном  $\varphi$ , обеспечивая либо заданное  $x_1$ , либо заданное  $x_2$ . Одновременное же достижение парой  $x_1, x_2$  заданных значений

невозможно, если только заданные значения  $x_1^*$  и  $x_2^*$  не согласованы.

В то же время, с практической точки зрения, больший интерес представляет задача приведения робота в заданную точку  $x_1, x_2$  за заданное время. По этой причине необходимо получить вариант управления роботом, где учитывается связь (9).

Для этого модифицируем задачу управления, назначив вместо (5) следующую критериальную функцию

$$\begin{aligned} J(x(t)) &= F_1(x_1(t) - x_1^*)^2 + \\ &+ F_3 \left( x_3(t) - \operatorname{arctg} \frac{x_2^*(t) - x_2(t)}{x_1^* - x_1(t)} \right)^2. \end{aligned} \quad (11)$$

Данная функция отражает следующую цель управления: приведение координаты  $x_1$  к значению  $x_1^*$ , а координаты  $\varphi \equiv x_3$  — к такому значению, которое обеспечит значение  $x_2^*$  в силу соотношения (9). Использование в (11) разностей под знаком арктангенса объясняется тем, что траектория движения робота криволинейная, поэтому в каждый текущий момент времени угол подхода к цели переменный и отсчитывается от достигнутого состояния.

Модификация алгоритма (2) для случая критерия (11) сводится к новым выражениям для  $J(x(t))$  (выражение (11)) и выражениям для компонент вектора

$$\frac{\partial J(x)}{\partial x} = \left( \frac{\partial J(x)}{\partial x_1} \quad \frac{\partial J(x)}{\partial x_2} \quad \frac{\partial J(x)}{\partial x_3} \right).$$

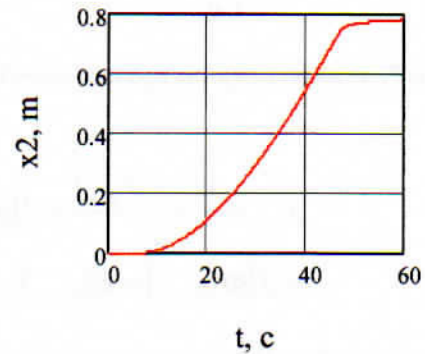
Последние теперь заметно сложнее, чем в случае критерия (5), особенно с учетом их реализации в бортовом компьютере робота, где в программном обеспечении нет обратных тригонометрических функций. Выражения для компонент имеют следующий вид (аргумент  $t$  для краткости записей опустим)

$$\frac{\partial J(x)}{\partial x_1} = 2F_1(x_1 - x_1^*) + \frac{2F_3(x_2 - x_2^*) \left( x_3 - \arctg \left( \frac{x_2 - x_2^*}{x_1 - x_1^*} \right) \right) x_2^*}{(x_1 - x_1^*)^2 \left( \frac{(x_2 - x_2^*)^2}{(x_1 - x_1^*)^2} + 1 \right)}, \quad (12)$$

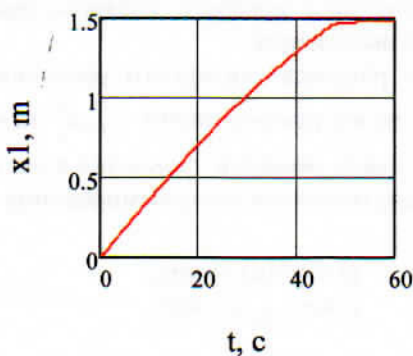
$$\frac{\partial J(x)}{\partial x_2} = \frac{2F_3 \left( x_3 - \arctg \left( \frac{x_2 - x_2^*}{x_1 - x_1^*} \right) \right) x_2^*}{(x_1 - x_1^*) \left( \frac{(x_2 - x_2^*)^2}{(x_1 - x_1^*)^2} + 1 \right)}, \quad (13)$$

$$\frac{\partial J(x)}{\partial x_3} = 2F_3 \left( x_3 - \arctg \left( \frac{x_2 - x_2^*}{x_1 - x_1^*} \right) \right). \quad (14)$$

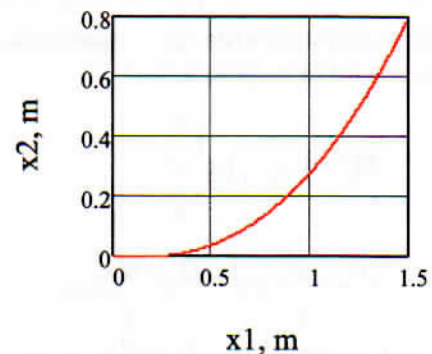
Управление (2) с критерием (11) проверено путем Mathcad- моделирования для случая  $x_1^* = 1.5$  м,  $x_2^* = 0.8$  м,  $t_f = 60$  с, шаге интегрирования  $h = 0,001$  с,  $F1 = 10$ ,  $F3 = 1$ , заданная точка достигнута с достаточно высокой точностью (1.488 м против 1.5 м для  $x_1$  и 0.784 м против 0.8 м для  $x_2$ ). В других проверенных случаях точность примерно такая же. На рисунках 1 – 5 показаны графики зависимостей координат, траектории и управляющих воздействий от времени.



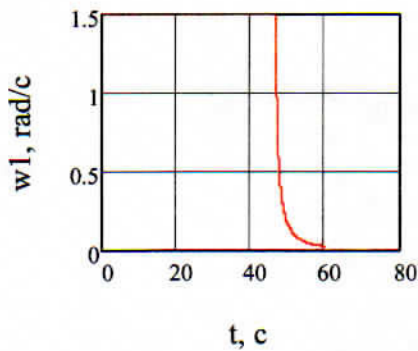
Р и с. 2. Зависимость от времени пути правого колеса



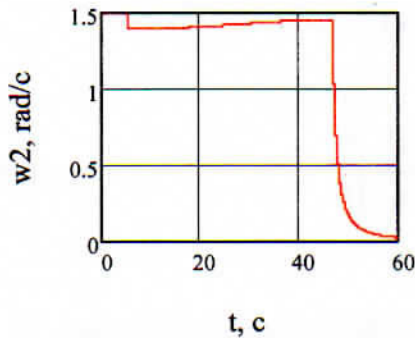
Р и с. 1. Зависимость от времени пути левого колеса



Р и с. 3. Траектория



Р и с. 4. Угловая скорость левого колеса



Р и с. 5. Угловая скорость правого колеса

**Дискретное управление.** Непрерывное управление роботом имеет тот недостаток, что для выполнения достаточно сложных расчетов по алгоритму (2), в особенности для случая критерия (11), требуется относительно много времени бортового компьютера. Так, для случая критерия (2) шаг интегрирования не может быть меньше 0,1 с. В то же время, текущие значения его координат, получаемых методом счисления по дискретизированной модели, должны определяться с возможно меньшим шагом. Это противоречие может быть устранено использованием управления, рассчитываемого с относительно большим шагом дискретности по времени.

Для расчета дискретного управления воспользуемся дискретным методом конечного состояния [3, 4]. В [3] приведены выводы расчетных выражений метода для дискретных и гибридных систем, а в [4] – для частного случая модели двухколесного робота. Дискретное управление рассчитывается путем решения нелинейного уравнения конечного типа относительно вектора управления

$$\begin{aligned}
 J_{j-1} + \frac{(J^* - J_{j-1})}{T_u} &= J(x_{j-1} + \Phi(x_{j-1}) + B(x_{j-1})u_j - x_{j-1}) = \\
 &= J(\Phi(x_{j-1}) + B(x_{j-1})u_j).
 \end{aligned}
 \tag{15}$$

В уравнении (15) верхними индексами обозначены номера компонент вектора состояния, нижними индексами – дискретные точки на оси времени в соответствии с тождествами  $u_j \equiv u(t_j)$ ,  $x_{j-1} \equiv x(t_{j-1})$ ,  $J_{j-1} \equiv J(t_{j-1})$ , вектор  $\Phi$  и матрица коэффициентов при управлении  $B$  определяются выражениями

$$\begin{aligned}
 \Phi(j-1, x_{j-1}) &= \begin{pmatrix} x_{j-1}^1 \\ x_{j-1}^2 \\ x_{j-1}^3 \end{pmatrix} \\
 B &= h_u \begin{pmatrix} \frac{D_w \cos(x_{j-1}^3)}{4} & \frac{D_w \cos(x_{j-1}^3)}{4} \\ \frac{D_w \sin(x_{j-1}^3)}{4} & \frac{D_w \sin(x_{j-1}^3)}{4} \\ \frac{D_w}{2L_r} & -\frac{D_w}{2L_r} \end{pmatrix},
 \end{aligned}$$

где  $h_u$  – шаг дискретизации расчета управления.

При этом левая часть (15) представляет собой правую часть конечно-разностного уравнения (дискретного аналога дифференциального уравнения (3)), решением которого является дискретная экспонента.

Для решения скалярного уравнения (15) с двумя неизвестными  $u_j^1, u_j^2$  приведем задачу решения уравнения к задаче математического программирования

$$\begin{aligned}
 Q = \varphi^2(u) &\rightarrow \min, \\
 U^{\min} &\leq u \leq U^{\max},
 \end{aligned}
 \tag{16}$$

где

$$\varphi(u) = J_{j-1} + \frac{(J^* - J_{j-1})}{T_u} - J(\Phi(x_{j-1}) + B(x_{j-1})u)$$

а  $U^{\min} = -\omega_m$ ,  $U^{\max} = \omega_m$  – векторы минимальных и максимальных значений искомых управлений.

Особенностью решения (15) на бортовом компьютере рассматриваемого робота является то, что в программном обеспечении робота нет процедур для решения систем уравнений или задач математического программирования типа задачи (16). Поэтому соответствующие процедуры необходимо разрабатывать самостоятельно. В [5] исследованы несколько методов решения задачи (16) с помощью известных методов математического программирования. Методы исследовались с помощью Matlab-моделирования и сравнивались со встроенным в Matlab методом Левенберга-Марквардта (эталонного метода). При этом для случая использования критерия (5) при управлении координатами  $x_1$  и  $\varphi$ , наибольшее быстродействие показал метод Ньютона (0,14 времени решения от времени решения эталонным методом). Неплохие результаты показал и конечно-разностный градиентный метод (0,29 времени решения от времени решения эталонным методом).

Рассмотрим возможности применения указанных выше двух методов решения уравнения (15) для случая управления координатами  $x_1$  и  $x_2$  посредством использования критерия (11).

Конечно-разностный метод Ньютона реализуется в виде рекуррентной формулы

$$u_k = u_{k-1} - (H_{k-1})^{-1} \nabla_{k-1}, \quad (17)$$

$$k = 1, 2, \dots$$

где  $\nabla_{k-1} = \frac{\partial Q}{\partial u} \Big|_{u=u_{k-1}}$  – градиент целевой

функции  $Q$ ,  $H_{k-1} = \frac{\partial^2 Q}{\partial u^2}$  – гессиан целевой функции. Номер  $k$  является номером итерации и не связан с номером дискретной точки времени. Эти обозначения введены с целью уменьшения количества индексов. При этом имеется в виду, что алгоритм (17) применяется многократно, в разных точках дискретного времени.

Конечно-разностный градиентный метод описывается рекуррентной формулой

$$u_k = u_{k-1} - \alpha_k \nabla_{k-1}, \quad k = 1, 2, \dots \quad (18)$$

где

$$\nabla = 2 \left( J_{j-1} + \frac{(J^* - J_{j-1})}{T_u} - J(x_j(u)) \right) \cdot \begin{pmatrix} \frac{\partial J(x_j(u))}{\partial x_j^1} & \frac{\partial J(x_j(u))}{\partial x_j^2} & \frac{\partial J(x_j(u))}{\partial x_j^3} \end{pmatrix} \cdot \begin{pmatrix} B_{11}(x_{j-1}) & B_{12}(x_{j-1}) \\ B_{21}(x_{j-1}) & B_{22}(x_{j-1}) \\ B_{31}(x_{j-1}) & B_{32}(x_{j-1}) \end{pmatrix},$$

$$J_{j-1} = F_1(x_{j-1}^1 - x_1^*)^2 + F_3 \left( x_{j-1}^3 - \arctg \left( \frac{x_2^* - x_{j-1}^2}{x_1^* - x_{j-1}^1} \right) \right)^2,$$

$$J(x_j) = F_1(x_j^1 - x_1^*)^2 + F_3 \left( x_j^3 - \arctg \left( \frac{x_2^* - x_j^2}{x_1^* - x_j^1} \right) \right)^2, \quad x_j(u) = x_{j-1} + B(x_{j-1})u,$$

$$H = \frac{\partial^2 Q}{\partial u^2} = \begin{pmatrix} \frac{\partial^2 Q}{\partial u^1 \partial u^1} & \frac{\partial^2 Q}{\partial u^1 \partial u^2} \\ \frac{\partial^2 Q}{\partial u^2 \partial u^1} & \frac{\partial^2 Q}{\partial u^2 \partial u^2} \end{pmatrix} = \begin{pmatrix} \frac{\partial \nabla^1}{\partial u^1} & \frac{\partial \nabla^1}{\partial u^2} \\ \frac{\partial \nabla^2}{\partial u^1} & \frac{\partial \nabla^2}{\partial u^2} \end{pmatrix},$$

$$\nabla^k = \frac{\partial Q}{\partial u^k} = -2 \left( J_{j-1} + \frac{(J^* - J_{j-1})}{T_u} - J(x_j(u)) \right) \cdot \sum_{i=1}^3 \frac{\partial J(x_j(u))}{\partial x_j^i} \cdot B_{ik}(x_{j-1}), \quad k = 1, 2$$

Как видим, решение задачи управления координатами  $x_1$  и  $x_2$  посредством использования критерия (11) заметно сложнее, чем при управлении координатами  $x_1$  и  $\varphi$  с критерием (5). Поэтому для случая критерия (11) нужны дополнительные исследования быстродействия методов решения уравнения (15). Может оказаться, что при этом градиентный метод окажется быстрее, чем метод Ньютона, ввиду существенного усложнения выражений для гессиана.

**Заключение.** В работе выполнены исследования по разработке и верификации алгоритмов управления мобильным роботом ArcBotics Sparki с целью приведения робота в точку с заданными координатами  $x_1^*$ ,  $x_2^*$ . Решение задачи получено при использовании непрерывного и дискретного управлений методом конечного состояния.

В случае непрерывного управления при наведении на точку  $x_1^* = 1.5$  м,  $x_2^* = 0.8$  м, за время  $t_f = 60$  с, достигнутая точность составляет величины 12 и 16 миллиметров соответственно (результаты получены Mathcad-моделированием).

Для случая дискретного управления использовано нелинейное уравнение конечного типа дискретного метода конечного состояния, для решения которого сформулирована задача нелинейного программирования. Получены выраже-

ния для градиента и гессиана целевой функции.

## СПИСОК ЛИТЕРАТУРЫ

1. *Подводный* робототехнический комплекс / Е.И. Есаулов, Е.Ю. Култыгин, В.Ф. Гуркин и др.: [Электронный ресурс] URL: <http://poleznayamodel.ru/model/10/102350.html>. (Дата обращения: 05.09.2016).

2. *Шушляпин Е.А.* Управление двухколесной платформой-носителем измерительных средств / Е.А. Шушляпин, А.Е. Безуглая // Системы контроля окружающей среды. Севастополь: ИПТС, 2016. Вып. 3(23). С.79–85.

3. *Шушляпин Е.А.* Управление нелинейными системами на основе прогноза конечного состояния неуправляемого движения. Севастополь: СевНТУ, 2012. 282 с.

4. *Афонина А.А.* Управление двухколесным роботом дискретным методом конечного состояния / А.А. Афонина, А.Е. Безуглая, Е.А. Шушляпин и др. // Наука и мир: Междунар.научн.журнал. 2016. № 4(32). С. 28–36.

5. *Афонина А.А.* Реализация дискретного управления двухколесным двухколейным роботом / А.А. Афонина, А.Е. Безуглая, Е.А. Шушляпин // Интеллектуальные системы, управление и мехатроника – 2016: Материалы Всероссийской научн.-техн. конф., Севастополь 19–21 мая 2016 г., Севастополь: СевГУ, 2016. С. 113–118.

## “ARCBOTICS SPARKI” MOBILE ROBOT CONTROL USING TERMINAL STATE METHOD

**E.A. Shushlyain, A.A. Afonina, A.E. Bezuglaya**

Federal State Educational Institution of Higher Education «Sevastopol State University»  
Russian Federation, Sevastopol, Universitetskaya St., 33

In this paper some investigations are made concerning development and verification of a two-wheeled mobile robot ArcBotics Sparki control algorithms bringing it to the point with given coordinates and any angle approach. The proposed criterion takes into account nonlinear kinematic relationship between the angle of approach and Cartesian coordinates of the robot current position. Developed control algorithms are based on continuous and discrete versions of the terminal state method.

**Key words:** two-track two-wheeled mobile robots, nonlinear systems, terminal control, input/output linearization method, terminal state method, mathematical programming.