

© 2024 г. Т.А. ТОРМАГОВ (tormagov@phystech.edu)  
(Московский физико-технический институт)

## МЕТОД ДЕФОРМАЦИИ ПУТЕЙ ОГРАНИЧЕННОЙ КРИВИЗНЫ ДЛЯ КОЛЕСНЫХ РОБОТОВ В ТОЧНОМ ЗЕМЛЕДЕЛИИ НА ОСНОВЕ КОНИЧЕСКОГО ПРОГРАММИРОВАНИЯ ВТОРОГО ПОРЯДКА

В точном земледелии является актуальной задача построения путей сельскохозяйственных роботов, покрывающих трехмерный участок ландшафта. Если для их реализации используются колесные роботы с рулением поворотом передних колес, то нормальная кривизна траекторий должна быть ограничена некоторой величиной, определяемой характеристиками этих машин. В работе рассмотрен метод деформации представленных однородными кубическими В-сплайнами путей для учета препятствий. Предложена оптимизационная задача, позволяющая производить расчет путей с минимизацией пропусков в покрытии. Данная задача является выпуклой и принадлежит к классу конического программирования второго порядка, что обуславливает возможность ее вычислительно эффективного решения. Приведены примеры вычислений.

*Ключевые слова:* точное земледелие, объезд препятствий, планирование путей, коническое программирование второго порядка, SOCP.

DOI: 10.31857/S0005231024020037, EDN: UKOYGH

### 1. Введение

В точном земледелии навигационные измерения и мониторинг с помощью различных сенсоров используются для совершенствования технологий выращивания сельскохозяйственных культур. Эффективное использование собранных данных позволяет, например, сократить время выполнения работ, площадь задействованных участков земли, количество необходимых ресурсов. Одной из задач планирования работ на поле в точном земледелии является построение путей сельскохозяйственных роботов, полностью покрывающих заданный участок трехмерного ландшафта.

Спланированные траектории движения должны быть реализуемы. Для роботов с механизмом руления поворотом передних колес для реализуемых траекторий должно выполняться условие на нормальную кривизну траектории  $u$

$$(1) \quad \|u\| < u_{\max},$$

где  $u_{\max}$  – максимальная реализуемая нормальная кривизна траектории. Значение  $u_{\max}$  определяется по формуле

$$(2) \quad u_{\max} = \frac{\operatorname{tg}\alpha_{\max}}{L},$$

где  $L$  – расстояние между передней и задней осями робота,  $\alpha_{\max}$  – максимальный эффективный угол поворота передних колес [1]. Для машины с двумя передними колесами углы поворота передних колес могут быть различны. По этой причине в формуле (2) используется эффективный угол поворота. Для сельскохозяйственной машины с дифференциальным приводом, допускающей разворот на месте, условие (1) не является необходимым для реализации траектории. В данной работе рассматривается планирование путей для машин с механизмом руления поворотом передних колес, для которых нормальная кривизна реализуемой траектории движения ограничена. Предполагается, что в процессе обработки поля движение производится передним ходом, а задний ход может использоваться только для разворотов на границе поля.

Построение путей, покрывающих заданный ландшафт, используется, например, при посеве и посадке растений, обработке их от вредителей, уборке урожая и стрижке газонов. Часто для покрытия поля используются параллельные или почти параллельные пути. Когда рельефом поля можно пренебречь, планирование путей производится для плоской поверхности. В [2] рассмотрены особенности построения плоских параллельных путей в пахотном земледелии, задача выбора направления построения путей с учетом формы и длины разворотов, задача декомпозиции поля сложной формы на участки, для каждого из которых планирование параллельных путей может быть проведено отдельно. Планирование путей по трехмерному ландшафту заметно отличается от плоского случая. В [3–5] построение трехмерного покрытия предлагается проводить относительно некоторого начального пути, пересекающего поле. В общем случае начальный путь является криволинейным. В [4] предложен поиск соседних путей с помощью решения задачи о пересечении цилиндра с поверхностью. В [6] предлагается подход к построению строго параллельных путей без ограничений на кривизну на криволинейной поверхности, заданной параметрически. Как показано в работе [3] по планированию трехмерных путей в пахотном земледелии, целевые функции, позволяющие оценивать покрытие поля, могут учитывать водную эрозию почвы, длины разворотов между рядами, пропуски в покрытии их-за ограничений на кривизну. В [5] предложен алгоритм построения путей по пологой поверхности с ограничениями на нормальную кривизну и на ширину перекрытия соседних полос.

Препятствия, находящиеся на поле, могут быть как подвижные (динамические), так и неподвижные (стационарные). Динамические препятствия меняют свое положение с течением времени. Примерами являются животные и другие сельскохозяйственные машины. Для автономных роботов траектории объезда таких препятствий планируются в процессе движения робота.

Информация о подвижных препятствиях в режиме реального времени собирается с помощью стереокамер, лидаров и других сенсоров [7]. Динамический учет препятствий требует перепланирования траектории [8]. Решение задач перепланирования траектории в реальном времени требует вычислительных ресурсов. По этой причине наличие на поле стационарных препятствий требует их учета при построении покрытия, до начала выполнения работ роботом. Отметим, что в любом из двух случаев при объезде препятствий машинами с рулением поворотом передних колес также должно выполняться условие (1).

В [2, 9–11] рассматриваются методы декомпозиции поля, позволяющие для различных конфигураций препятствий разделить его на отдельные участки, каждый из которых можно покрыть параллельными путями. Такое разделение, как правило, требует выделения достаточно широкой области для разворотов вблизи препятствия. В литературе рассмотрено множество задач, в которых траектории обхода препятствий строятся с минимизацией длины пути. Здесь можно выделить подходы, основанные на применении алгоритмов поиска на графе [12, 13], и алгоритмы деформации пути с применением штрафных функций (искусственных потенциальных полей) [14–16]. Однако в точном земледелии решение задачи деформации пути с минимизацией его длины может привести к значительным пропускам в покрытии. В связи с этим требуется подход, позволяющий сокращать площадь необрабатываемой области.

В данной работе приведен метод, позволяющий для учета препятствий проводить перепланирование покрытия поля путями с помощью решения задачи конического программирования второго порядка для машин с механизмом руления поворотов передних колес. Задачи конического программирования второго порядка (англ. second-order cone programming, SOCP) в общем виде можно сформулировать следующим образом [17, 18].

*Задача 1. Найти*

$$(3) \quad \min_x f^T x$$

*при ограничениях*

$$(4) \quad \|A_i x + b_i\| \leq c_i^T x + d_i, \quad i = 1, \dots, N,$$

где  $x \in R^l$  ( $l$  – размерность переменной),  $f \in R^l$ ,  $A_i \in R^{(l_i-1) \times l}$ ,  $b_i \in R^{l_i-1}$ ,  $c_i \in R^l$ ,  $d_i \in R$ ,  $N$  – число ограничений,  $l_i$  – натуральные числа.

Здесь и далее обозначение  $\|\cdot\|$  будем использовать для евклидовых норм векторов. Задачи SOCP являются выпуклыми и допускают вычислительно эффективные методы решения, см. [19, 20].

Отметим, что после построения покрытия поля путями нужно определить маршрут движения одной или нескольких машин по полученным рядам, позволяющий провести обработку за минимальное время либо с минимальной длиной маршрута. В общем виде маршрутизация является вычислительно трудоемкой задачей. Ее точное решение часто не может быть получено за

приемлемое время. Поэтому на практике часто используются эвристические алгоритмы маршрутизации. Примеры таких алгоритмов, основанных на методе имитации отжига, приведены в [5, 21, 22].

## 2. Предлагаемый метод

Предположим, что покрытие поля путями построено без учета препятствий. Требуется изменить пути, пересекающие границы препятствий, так, чтобы каждый путь был реализуем, а площадь необработанных участков была как можно меньшей. Для реализуемости нужно, чтобы путь не пересекал препятствия и выполнялось условие на кривизну (1).

Для задания координат точек пространства для данного метода удобно использовать локальную правую декартову систему координат ENU ( $x, y, z$ ) (англ. *East, North and Up*). Для ее определения фиксируется некоторая точка на поле, которая далее используется как начало координат системы. Ось  $x$  направляют на восток, ось  $y$  – на север, а ось  $z$  дополняет набор координатных осей до правой тройки. При этом оси  $x$  и  $y$  лежат в касательной плоскости к используемому эллипсоиду модели Земли. Для описания поверхности, по которой движутся роботы, используется цифровая модель высот (англ. *Digital Elevation Model, DEM*). Она задается функцией  $z = z(x, y)$ , позволяющей определить для известных координат  $x$  и  $y$  высоту  $z$ . При этом на практике в большинстве случаев точное определение высоты для каждой точки поля невозможно и используется интерполяция высот. Для этого предварительно собираются данные о высотах в некоторых (опорных) точках на поле. Затем по полученным данным строится функция определения высоты  $z = z(x, y)$ . Примерами методов построения таких функций для данных на равномерной сетке являются билинейная и бикубическая интерполяции [4], а по неравномерной сетке – двумерная В-сплайновая интерполяция [23] и кригинг [24].

Для описания траектории движения колесных роботов удобно использовать гладкие функции. Примером являются однородные кубические В-сплайны [25]. Сплайновая кривая определяется набором контрольных точек  $r_1, r_2, \dots, r_n$ . Набор дополняется точками

$$(5) \quad r_0 = 2r_1 - r_2, \quad r_{n+1} = 2r_n - r_{n-1}.$$

Каждые четыре соседние контрольные точки  $r_{i-1}, r_i, r_{i+1}, r_{i+2}$  задают элементарный В-сплайн  $r^{(i)}(t)$ ,  $i = 1, \dots, n - 1$ , параметрической формулой (рис. 1)

$$(6) \quad r^{(i)}(t) = \frac{(1-t)^3}{6}r_{i-1} + \frac{4-6t^2+3t^3}{6}r_i + \frac{1+3t+3t^2-3t^3}{6}r_{i+1} + \frac{t^3}{6}r_{i+2},$$

где параметр сплайна  $t$  принимает значения из отрезка  $[0, 1]$ . Кривая  $r(t)$ ,  $t \in [0, n]$  состоит из набора элементарных сплайнов  $r^{(i)}(t)$ ,  $i = 1, \dots, n$ , и может быть определена как

$$(7) \quad r(t) = r^{(i)}(t - (i - 1)), \quad i = \lfloor t + 1 \rfloor, \quad t \in [0; n),$$

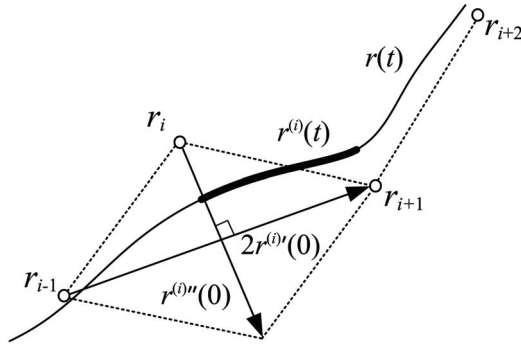


Рис. 1. Сплайнная кривая  $r(t)$ , элементарный сплайн  $r^{(i)}(t)$  (выделен на сплайнной кривой), определяемый по четырем эквидистантным контрольным точкам  $r_{i-1}$ ,  $r_i$ ,  $r_{i+1}$ ,  $r_{i+2}$ , векторы  $r^{(i)'}(0)$  и  $r^{(i)''}(0)$ .

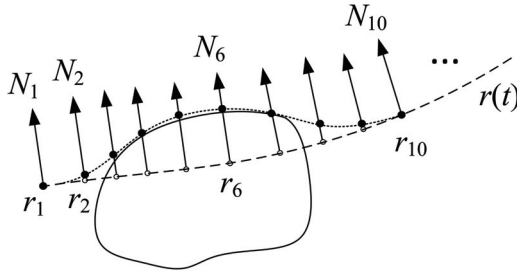


Рис. 2. Контур препятствия (сплошная линия), сплайнная кривая  $r(t)$  до (штриховая линия) и после (пунктирная линия) деформации пути, направления поиска  $N_i$  и контрольные точки  $r_i$ .

$r(n) = r^{(n-1)}(1)$ , где  $\lfloor t \rfloor$  означает целую часть (округление вниз) для параметра  $t$ . Как видно из формул (6) и (7),  $r(t)$  является непрерывной с непрерывной первой и второй производной по параметру, в том числе в местах стыка элементарных сплайнов. Отметим, что и сплайнная траектория, и контрольные точки могут не лежать на поверхности  $z = (x, y)$ , определяемой цифровой моделью высот. Однако они будут находиться вблизи этой поверхности, что пригодно для практического применения такого описания путей.

Метод построения покрытия поля без препятствий путями ограниченной кривизны при описании траекторий однородными кубическими В-сплайнами изложен в [5, 7]. Наличие на поле препятствий не позволяет использовать эти построения напрямую. Для этого случая в данной работе предлагается модифицировать покрытие поля путями, построенное без учета препятствий, с помощью решения набора задач деформации пути. Поскольку сплайнная кривая задается набором контрольных точек, то определение деформированного пути сводится к заданию его контрольных точек.

Для координат контрольных точек воспользуемся обозначениями  $x_i, y_i, z_i$ , такими, что  $r_i = (x_i, y_i, z_i)$ . Определим набор векторов единичной длины  $N_i = (N_i^x, N_i^y, N_i^z) \in R^3$ ,  $i = 1, \dots, n$ , так, что эти векторы параллельны кас-

тельными плоскостями к поверхности в точках  $(x_i, y_i, z(x_i, y_i))$  (либо лежат в этих плоскостях); при этом направление нормали проекции сплайна на касательную плоскость для  $(x_i, y_i, z(x_i, y_i))$  совпадает с  $N_i$ . Вдоль этих направлений  $N_i$  будем искать новые позиции контрольных точек сплайна  $r_i + N_i d_i$ ,  $i = 1, \dots, n$ , где диапазоны изменения значений  $d_i$  определяются геометрическими ограничениями задачи (рис. 2).

Рассмотрим отдельно учет ограничений на нормальную кривизну траектории (1).

*Утверждение 1 (достаточные условия). Если для плоского однородного кубического B-сплайна с контрольными точками  $r_i + N_i d_i$ ,  $i = 1, \dots, n$ , и дополнением (5) выполнены условия*

$$(8) \quad \left\| \begin{array}{l} x_{i-1} + N_{i-1}^x d_{i-1} - 2(x_i + N_i^x d_i) + x_{i+1} + N_{i+1}^x d_{i+1} \\ y_{i-1} + N_{i-1}^y d_{i-1} - 2(y_i + N_i^y d_i) + y_{i+1} + N_{i+1}^y d_{i+1} \end{array} \right\| \leq u_{\max} \hat{l}_i(d_{i-1}, d_{i+1}),$$

$$(9) \quad 4\hat{l}_i(d_{i-1}, d_{i+1}) = (x_{i+1} - x_{i-1})^2 + 2(x_{i+1} - x_{i-1})(N_{i+1}^x d_{i+1} - N_{i-1}^x d_{i-1}) + \\ + (y_{i+1} - y_{i-1})^2 + 2(y_{i+1} - y_{i-1})(N_{i+1}^y d_{i+1} - N_{i-1}^y d_{i-1}),$$

где  $i = 1, 2, \dots, n$ , то (1) выполняется для точек стыка элементарных сплайнов.

Отметим, что из (8) следует выполнение условия (1) лишь для точек стыка элементарных сплайнов. Пригодность этого условия для практических применений обусловлена двумя факторами. Первый заключается в том, что максимальное значение  $\|r^{(i)''}(t)\|$  для каждого элементарного сплайна достигается при  $t = 0$  или  $t = 1$  (доказательство приведено в [5]). Второй заключается в том, что  $\|r^{(i)'}(t)\|$  отражает соответствие между малыми изменениями длины траектории и соответствующими изменениями параметра сплайна и для элементарного сплайна с эквидистантными контрольными точками является примерно постоянной величиной.

В случае, когда поверхность плоская, но не параллельна и не совпадает с плоскостью  $(x, y)$ , условия на нормальную кривизну могут быть получены из (8) поворотом системы координат. В случае криволинейной поверхности вектор кривизны  $k(t)$  может не совпадать с вектором нормальной кривизны  $u(t)$ . Рассмотрим случай достаточно пологих поверхностей, для которых движение на расстояние  $2D_s$ , где  $D_s$  – максимальная длина вектора между соседними контрольными точками сплайнов, можно с хорошей точностью считать движением в наклонной плоскости. В точном земледелии для большого числа приложений такое предположение допустимо. Значение модуля вектора  $u(t)$  в таком случае можно аппроксимировать значением кривизны траектории в касательной плоскости. Определим вектор  $r'_i$  как

$$(10) \quad r'_i = \frac{r_{i+1} - r_{i-1}}{\|r_{i+1} - r_{i-1}\|}.$$

Для каждой контрольной точки  $r_i$  введем плоскую систему координат  $(\tilde{x}, \tilde{y})$  с началом в точке  $r_i$  с координатными осями, сонаправленными векторам  $r'_i$  и  $N_i$  соответственно. В новых системах координат для контрольной точки  $r_i$  условия (8) принимают вид

$$(11) \quad \left\| \begin{array}{l} \tilde{x}_{i-1} + \tilde{N}_{i-1}^x d_{i-1} + \tilde{x}_{i+1} + \tilde{N}_{i+1}^x d_{i+1} \\ \tilde{y}_{i-1} + \tilde{N}_{i-1}^y d_{i-1} - 2d_i + \tilde{y}_{i+1} + \tilde{N}_{i+1}^y d_{i+1} \end{array} \right\| \leq u_{\max} \tilde{l}_i(d_{i-1}, d_{i+1}),$$

где

$$(12) \quad 4\tilde{l}_i(d_{i-1}, d_{i+1}) = (\tilde{x}_{i+1} - \tilde{x}_{i-1})^2 + 2(\tilde{x}_{i+1} - \tilde{x}_{i-1})(\tilde{N}_{i+1}^x d_{i+1} - \tilde{N}_{i-1}^x d_{i-1}) + \\ + (\tilde{y}_{i+1} - \tilde{y}_{i-1})^2 + 2(\tilde{y}_{i+1} - \tilde{y}_{i-1})(\tilde{N}_{i+1}^y d_{i+1} - \tilde{N}_{i-1}^y d_{i-1}).$$

Здесь  $(\tilde{x}_{i-1}, \tilde{y}_{i-1})$  и  $(\tilde{x}_{i+1}, \tilde{y}_{i+1})$  – координаты проекций соседних контрольных точек в системе  $(\tilde{x}, \tilde{y})$ , а  $(\tilde{N}_{i-1}^x, \tilde{N}_{i-1}^y)$  и  $(\tilde{N}_{i+1}^x, \tilde{N}_{i+1}^y)$  – координаты векторов их нормалей в системе  $(\tilde{x}, \tilde{y})$ . Отметим, что система координат  $(\tilde{x}, \tilde{y})$  в общем виде различна для каждого  $r_i$ , а значит, результаты вычислений  $(\tilde{x}_{i-1}, \tilde{y}_{i-1})$ ,  $(\tilde{x}_{i+1}, \tilde{y}_{i+1})$ ,  $(\tilde{N}_{i-1}^x, \tilde{N}_{i-1}^y)$ ,  $(\tilde{N}_{i+1}^x, \tilde{N}_{i+1}^y)$  для  $r_i$  нельзя использовать при записи условий на кривизну для смещения точек  $r_{i-2}$  и  $r_{i+2}$  (если они существуют). Для этих точек требуется отдельное вычисление проекций в системах координат с началами в  $r_{i-2}$  и в  $r_{i+2}$  соответственно.

Условия на кривизну (11) имеют вид конусов второго порядка (4). Правая часть этих условий линейно зависит от значений переменных  $d_i$ ,  $i = 1, \dots, n$ . Это позволяет сформулировать задачу деформации пути с ограничением на кривизну в виде следующей задачи конического программирования второго порядка.

*Задача 2. Найти*

$$(13) \quad \min_q q_{n+1}$$

*при ограничении (11) для  $i = 1, \dots, n$ ,*

$$(14) \quad \|w^T q\| \leq q_{n+1},$$

$$(15) \quad \overline{d}_i \leq q_i \leq \overline{\overline{d}}_i, \quad i = 1, \dots, n,$$

*где  $q \in R^{n+1}$ ,  $q_i$  –  $i$ -я компонента вектора  $q$ .*

Для оптимального решения задачи первые  $n$  компонент вектора  $q$  равны смещениям контрольных точек, которые требуется провести:  $d_i = q_i$ ,  $i = 1, \dots, n$ . Значения  $\overline{d}_i$  и  $\overline{\overline{d}}_i$ ,  $i = 1, \dots, n$  определяют геометрические ограничения на отклонения контрольных точек вдоль направления  $N_i$ , обусловленные препятствиями и границами поля. При этом для каждого препятствия выбирается направление обхода. Первые  $n$  компонент вектора  $w \in R^{n+1}$  являются фиксированными значениями весовой функции, которые могут быть

выбраны как одинаковыми, так и зависящими от расположения контрольных точек относительно препятствий. Последняя компонента вектора  $w$  равна нулю:  $w_{n+1} = 0$ . Критерий оптимизации (13) в сочетании с условием (14) позволяет минимизировать смещение контрольных точек и сократить пропуски в покрытии, образующиеся при объезде препятствий из-за ограничения на нормальную кривизну пути.

На практике для определения значений  $\overline{d}_i$  и  $\overline{\overline{d}}_i$ ,  $i = 1, \dots, n$  можно использовать представление границ препятствий и контура поля замкнутыми ломаными линиями, проекции которых на плоскость локального горизонта будут представлять собой многоугольники. В плоскости локального горизонта поиск  $\overline{d}_i$  и  $\overline{\overline{d}}_i$  сводится к выбору направления обхода каждого препятствия и решению задачи о пересечении прямой и многоугольника.

Приведенная задача оптимизации является выпуклой и допускает вычислительно эффективные методы решения. Другой вычислительной особенностью таких задач является то, что вывод о совместности или несовместности ограничений может быть сделан в процессе решения. Отметим, что предлагаемый подход не нацелен на поиск направлений обхода препятствий, а служит инструментом построения путей в заданной выпуклой области, определенной ограничениями (15). Если задача деформации пути оказалась несовместной, а допустимый путь в заданных границах существует, то это может быть обусловлено невозможностью выполнения условий на кривизну при смещении точек вдоль выбранных направлений. Невозможность выполнения ограничений на кривизну может быть обусловлена тем, что при больших смещениях нарушается равномерность расположения контрольных точек. Кроме того, проекции на плоскость  $(x, y)$  направлений поиска  $N_i$  могут пересекаться, что приводит к решениям с самопересекающимися траекториями, для которых, как правило, нарушается условие на кривизну. Перечисленные проблемы можно устранить применением следующего алгоритма деформации пути, основанного на ослаблении ограничений задачи с последующим представлением пути равномерно расположенными контрольными точками.

*Алгоритм 1* (деформации пути).

1. Решается задача 2 при исходном ограничении  $u_{\max}$ . Если решение найдено, то производится перепараметризация траектории равномерно расположенными контрольными точками. Задача 2 решается повторно для текущего расположения контрольных точек. Если решение найдено, то оно возвращается в качестве ответа.

2. Решается задача 2 для текущего расположения контрольных точек для максимального значения нормальной кривизны  $u_{\max} + \Delta$ ,  $\Delta > 0$ . Если оптимальное решение найдено, то производится перепараметризация траектории равномерно расположенными контрольными точками и выполняется возврат на шаг 1. Если не найдено, то значение  $\Delta$  увеличивается, после чего повторяются действия шага 2.



Алгоритм на каждом этапе его работы предполагает выбор изменения кривизны  $\Delta$ . От того, как этот выбор произведен, зависит, сколько итераций алгоритма потребуется выполнить, прежде чем решение задачи будет найдено.

### 3. Примеры вычислений

На рис. 3 показан пример решения задачи деформации путей для трех препятствий,  $u_{\max} = 0,4 \text{ м}^{-1}$ , ширина рядов 1 м. Покрытие поля путями сначала построено без учета препятствий с помощью метода, приведенного в [5], а затем произведена деформация пути с единичными весами ( $w_i = 1$ ) для всех контрольных точек. Параметр  $\Delta$  выбирался равным

$$(16) \quad \Delta = (u_{\max}^{-1} - 0,09m)^{-1} - u_{\max},$$

где  $m$  – номер итерации алгоритма деформации пути на шаге 2. Такая релаксация условия на кривизну соответствует уменьшению допустимого мгновенного радиуса кривизны траектории на 0,09 м за каждую итерацию. Очевидно, что число итераций при этом ограничено: необходимо обеспечить неотрицательное значение выражения  $u_{\max}^{-1} - 0,09m$ .

На рис. 4 приведен пример деформации путей с применением метода штрафных функций (искусственных потенциальных полей [15, 16]), а на рис. 5 – результат применения рассматриваемого в данной работе метода деформации путей на основе решения задачи 2 (SOCP). Видно, что крите-

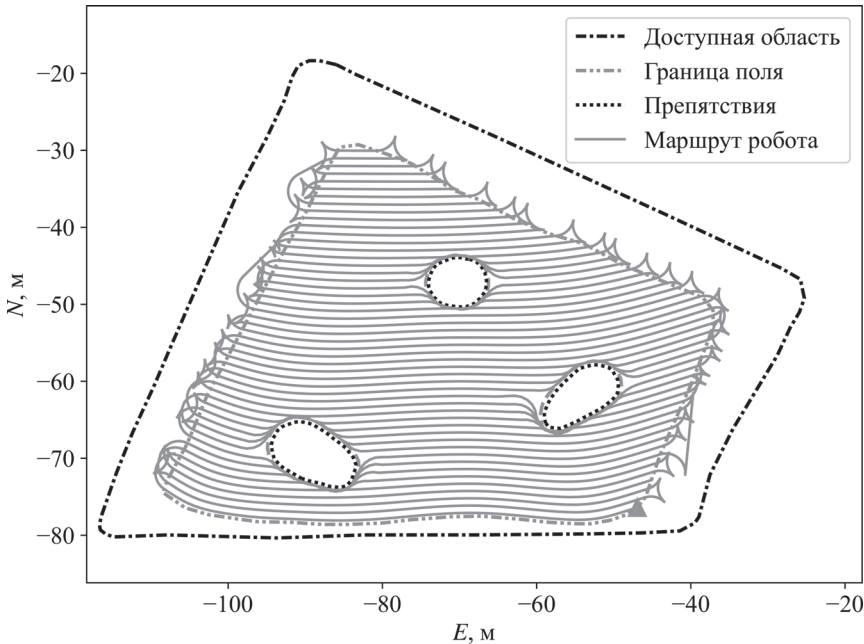


Рис. 3. Пример деформации путей с тремя препятствиями на поле.

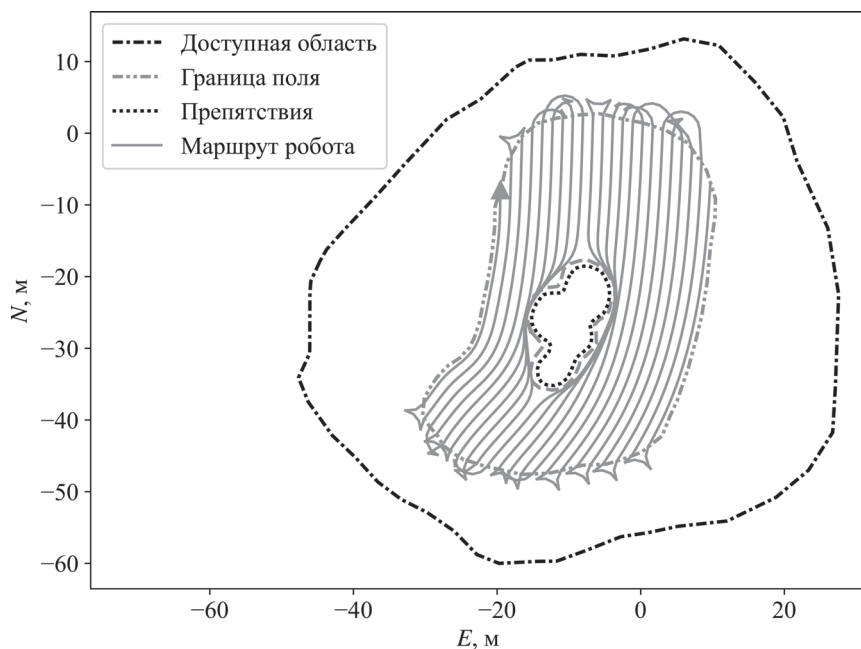


Рис. 4. Пример деформации путей с применением метода штрафных функций.

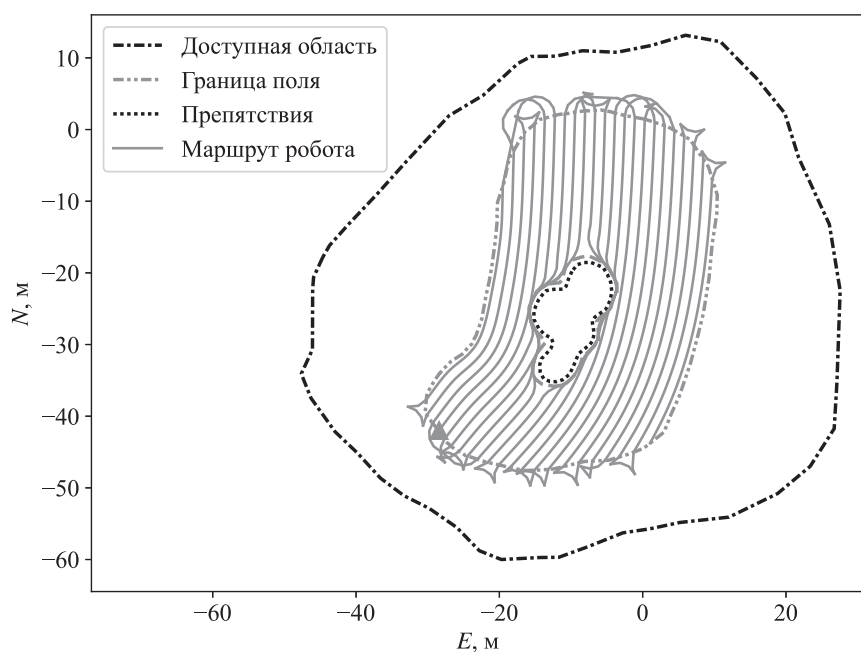


Рис. 5. Пример деформации путей на основе решения задачи SOCP.

рий оптимизации (13) задачи 2 с условием (14) позволил сократить необработанную область вокруг невыпуклого препятствия, образующуюся из-за учета ограничений на нормальную кривизну.

Для всех примеров для задач конического программирования второго порядка использован алгоритм решения, описанный в [19], расчет производился с использованием CVXPY [26, 27]. Маршруты обхода рядов построены для одного робота с разворотами передним и задним ходом. Для оптимизации длины маршрута использован алгоритм муравьиных колоний (англ. ant colony optimization, ACO), особенности применения которого к задачам маршрутизации рассмотрены в [28]. Отметим, что минимизация длины маршрута для данного примера не эквивалентна минимизации времени работ, по крайней мере из-за необходимости остановок при переключении между передним и задним ходом на разворотах.

#### 4. Заключение

В работе предложен метод деформации траекторий для учета препятствий, применимый в точном земледелии при планировании покрытия поля путями. Результаты получены для случая описания траектории движения однородными кубическими B-сплайнами. Данный метод позволяет получить пути для колесных роботов с механизмом руления поворотом передних колес, максимальная реализуемая нормальная кривизна пути для которых ограничена. При этом построение траекторий производится с минимизацией евклидовой нормы взвешенного вектора отклонений контрольных точек сплайнов начальных путей, построенных без учета препятствий, что позволяет сократить площадь необработанной области. Научная новизна данного подхода заключается в формулировке деформации пути с ограничением на кривизну в виде задачи конического программирования второго порядка, которая является выпуклой и допускает вычислительно эффективные методы решения. Ограничением рассматриваемого подхода является требование достаточной пологости рельефа поля, которое при практических применениях в точном земледелии часто выполняется.

#### ПРИЛОЖЕНИЕ

*Доказательство утверждения 1.* В случае движения в плоскости  $(x, y)$  модуль вектора кривизны  $k(t)$  сплайновой кривой в точке  $r^{(i)}(t)$  описывается формулой

$$(П.1) \quad \|k(t)\| = \frac{\|r^{(i)''}(t)\| \sin \varphi(t)}{\|r^{(i)'}(t)\|^2},$$

где  $r^{(i)'}(t)$  и  $r^{(i)''}(t)$  – векторы первой и второй производной сплайна по параметру,  $\varphi(t)$  – угол между векторами  $r^{(i)'}(t)$  и  $r^{(i)''}(t)$ . Вектор кривизны кривой  $k(t)$  в таком случае совпадает с вектором нормальной кривизны  $u(t)$ . Векторы первой и второй производных в точке стыка элементарных сплайнов

ИМЕЮТ ВИД

$$(II.2) \quad r^{(i)'}(0) = \frac{1}{2}(r_{i+1} + N_{i+1}d_{i+1} - r_{i-1} - N_{i-1}d_{i-1}),$$

$$(II.3) \quad r^{(i)''}(0) = r_{i-1} + N_{i-1}^r d_{i-1} - 2(r_i + N_i^r d_i) + r_{i+1} + N_{i+1}^r d_{i+1}.$$

Заметим, что функция  $\hat{l}_i(d_{i-1}, d_{i+1})$ , определяемая формулой (9), является оценкой снизу для  $\|r^{(i)'}(0)\|^2$ , не учитывающей квадратичные слагаемые по смещению:

$$(II.4) \quad \begin{aligned} \|r^{(i)'}(0)\|^2 &= \hat{l}_i(d_{i-1}, d_{i+1}) + \frac{1}{4}(N_{i+1}^x d_{i+1} - N_{i-1}^x d_{i-1})^2 + \\ &+ \frac{1}{4}(N_{i+1}^y d_{i+1} - N_{i-1}^y d_{i-1})^2. \end{aligned}$$

Тогда из (8) следует, что

$$(II.5) \quad \left\| r^{(i)''}(0) \right\| \sin \varphi(0) \leq u_{\max} \left\| r^{(i)'}(0) \right\|^2,$$

а значит, в точках стыка элементарных сплайнов условие (1) выполнено.

#### СПИСОК ЛИТЕРАТУРЫ

1. *Gilimyanov R.F., Pesterev A.V., Rapoport L.B.* Smoothing curvature of trajectories constructed by noisy measurements in path planning problems for wheeled robots // *J. Comput. Syst. Sci. Int.* 2008. V. 47. No. 5. P. 812–819.
  2. *Jin J., Tang L.* Optimal Coverage Path Planning for Arable Farming on 2D Surfaces // *Trans. ASABE*. St. Joseph, MI: ASABE, 2010. V. 53. No. 1. P. 283–295.
  3. *Jin J., Tang L.* Coverage path planning on three-dimensional terrain for arable farming // *J. F. Robot.* 2011. V. 28. No. 3. P. 424–440.
  4. *Hameed I.A., La Cour-Harbo A., Osen O.L.* Side-to-side 3D coverage path planning approach for agricultural robots to minimize skip/overlap areas between swaths // *Rob. Auton. Syst.* Elsevier. 2016. Vol. 76. P. 36–45.
  5. *Tormagov T., Rapoport L.* Coverage Path Planning for 3D Terrain with Constraints on Trajectory Curvature Based on Second-Order Cone Programming / *Advances in Optimization and Applications* Ed. Olenev N.N. et al. Cham: Springer International Publishing, 2021. P. 258–272.
  6. *Galvez A., Iglesias A., Puig-Pey J.* Computing parallel curves on parametric surfaces // *Appl. Math. Model.* 2014. V. 38. No. 9–10. P. 2398–2413.
  7. *Тормагов Т.А., Генералов А.А., Шавин М.Ю., Рапопорт Л.Б.* Задачи управления движением автономных колесных роботов в точном земледелии // *Гироскопия и навигация*. 2022. Т. 30, 1 (116). С. 39–60.
- Tormagov T.A., Generalov A.A., Shavin M.Y., Rapoport L.B.* Motion Control of Autonomous Wheeled Robots in Precision Agriculture // *Gyroscopy Navig.* 2022. V. 13. No. 1. P. 23–35.

8. *Chichkanov I., Shawin M.* Algorithm for Finding the Optimal Obstacle Avoidance Maneuver for Wheeled Robot Moving Along Trajectory // 2022 16th International Conference on Stability and Oscillations of Nonlinear Control Systems (Pyatnitskiy's Conference). Moscow: IEEE, 2022. P. 1–3.
9. *Latombe J.-C.* Robot Motion Planning. Boston, MA: Springer US, 1991.
10. *Choset H., Pignon P.* Coverage Path Planning: The Boustrophedon Cellular Decomposition / Field and Service Robotics Ed. Zelinsky A. London: Springer London, 1998. P. 203–209.
11. *Acar E.U., Choset H., Rizzi A.A. et al.* Morse Decompositions for Coverage Tasks // Int. J. Rob. Res. 2002. V. 21. No. 4. P. 331–344.
12. *Hart P., Nilsson N., Raphael B.* A Formal Basis for the Heuristic Determination of Minimum Cost Paths // IEEE Trans. Syst. Sci. Cybern. 1968. V. 4. No. 2. P. 100–107.
13. *Stentz A.* Optimal and efficient path planning for unknown and dynamic environments // Int. J. Robot. Autom. Int Association of Science and Technology for Development, 1995. V. 10. No. 3. P. 89–100.
14. *Chuang J.-H.* Potential-based modeling of three-dimensional workspace for obstacle avoidance // Proceedings IEEE International Conference on Robotics and Automation. IEEE Comput. Soc. Press. 1993. P. 19–24.
15. *Гилмьянов Р.Ф., Рапопорт Л.Б.* Метод деформации пути в задачах планирования движения роботов при наличии препятствий // Проблемы управления. 2012. № 1. С. 70–76.
16. *Gilimyanov R.F., Rapoport L.B.* Path Deformation Method for Robot Motion Planning Problems in the Presence of Obstacles // Autom. Remote Control. 2013. V. 74. No. 12. P. 70–76.
17. *Lobo M.S., Vandenberghe L., Boyd S. et al.* Applications of second-order cone programming // Linear Algebra Appl. North-Holland, 1998. V. 284. No. 1–3. P. 193–228.
18. *Boyd S., Vandenberghe L.* Convex Optimization. Cambridge: Cambridge University Press, 2004.
19. *O'Donoghue B., Chu E., Parikh N. et al.* Conic Optimization via Operator Splitting and Homogeneous Self-Dual Embedding // J. Optim. Theory Appl. 2016. V. 169. No. 3. P. 1042–1068.
20. *Domahidi A., Chu E., Boyd S.* ECOS: An SOCP solver for embedded systems // 2013 European Control Conference. 2013. P. 3071–3076.
21. *Vahdanjoo M., Zhou K., Sorensen C.A.G.* Route Planning for Agricultural Machines with Multiple Depots: Manure Application Case Study // Agronomy. 2020. V. 10, No. 10. P. 1608.
22. *Conesa-Munoz J., Bengochea-Guevara J., Andujar D., et al.* Route planning for agricultural tasks: A general approach for fleets of autonomous vehicles in site-specific herbicide applications // Comput. Electron. Agric. Elsevier B.V., 2016. V. 127. P. 204–220.
23. *Dierckx P.* An Algorithm for Surface-Fitting with Spline Functions // IMA J. Numer. Anal. 1981. V. 1, No. 3. P. 267–283.
24. *Cressie N.* The origins of kriging // Math. Geol. Kluwer Academic Publishers-Plenum Publishers, 1990. V. 22, No. 3. P. 239–252.

25. *Пестерев А.В., Гиллмьянов Р.Ф.* Планирование пути для колесного робота // Тр. ИСА РАН. 2006. № 25. С. 205–212.
26. *Diamond S., Boyd S.* CVXPY: A Python-embedded modeling language for convex optimization // J. Mach. Learn. Res. 2016. V. 17, No. 83. P. 1–5.
27. *Agrawal A., Verschueren R., Diamond S. et al.* A rewriting system for convex optimization problems // J. Control Decis. 2018. V. 5, No. 1. P. 42–60.
28. *Junjie P., Dingwei W.* An Ant Colony Optimization Algorithm for Multiple Travelling Salesman Problem // First International Conference on Innovative Computing, Information and Control – Volume I (ICICIC'06). IEEE. 2006. V. 1. P. 210–213.

*Статья представлена к публикации членом редколлегии Л.Б. Рапопортом.*

Поступила в редакцию 26.06.2023

После доработки 30.12.2023

Принята к публикации 20.01.2024