

УДК 519.71

# ПРИМЕНЕНИЕ УПРАВЛЕНИЯ С ПРОГНОЗИРУЮЩИМИ МОДЕЛЯМИ И СТОХАСТИЧЕСКОЙ ОПТИМИЗАЦИЕЙ В ЗАДАЧЕ ДЕЦЕНТРАЛИЗОВАННОГО МНОГО-АГЕНТНОГО ИЗБЕГАНИЯ СТОЛКНОВЕНИЙ

**С.А. Дергачев**

*Национальный исследовательский университет «Высшая школа экономики»  
Федеральный исследовательский центр «Информатика и управление» Российской Академии Наук*  
Россия, 109028, Москва, Покровский бульвар, 11,  
Россия, 119333, Москва, Вавилова, 44, кор.2  
E-mail: dergachev@isa.ru

**К.С. Яковлев**

*Федеральный исследовательский центр «Информатика и управление» Российской Академии Наук  
Национальный исследовательский университет «Высшая школа экономики»*  
Россия, 119333, Москва, Вавилова, 44, кор.2,  
Россия, 109028, Москва, Покровский бульвар, 11  
E-mail: yakovlev@isa.ru

**Ключевые слова:** избегание столкновений, кинематические ограничения, децентрализованная много-агентная навигация, Model Predictive Path Integral.

**Аннотация:** Избегание столкновений является важнейшим компонентом децентрализованной много-агентной системы навигации. Большинство существующих много-агентных методов избегания столкновений либо не учитывают кинематических ограничений, либо адаптированы лишь к определенным моделям движения. В рамках этой работы предлагается новый подход к избеганию столкновений, позволяющий находить безопасные управления для аффинных динамических систем. Предлагаемый подход основан на стохастическом методе *MPPI* и линейных ограничениях, полученных алгоритмом *ORCA*. Результаты экспериментального сравнения предлагаемого подхода с современными методами избегания столкновений показывают, что во многих случаях он превосходит существующие аналоги с точки зрения скорости достижения целей группой агентов.

## 1. Введение

Много-агентная навигация – сложная задача, возникающая в различных областях, таких как мобильная робототехника, разработка видеоигр и моделирование поведения толпы. В случае, если устойчивая связь между агентами не гарантирована, то такая задача может быть решена только децентрализованным способом.

Важнейшим аспектом децентрализованных навигационных систем является компонент избегания столкновений с другими агентами при движении к целевой позиции. Безопасное движение к цели достигается путем итеративного выбора управляющих воздействий каждым агентом. Одновременно, важно учитывать как можно больше факторов, включая кинематические ограничения агента. Однако существующие методы либо полностью игнорируют такие ограничения (например, [1]), либо фокусируются на конкретных динамических моделях (например, [2]).

В этой работе представляется алгоритм избегания столкновений, разработанный для широкого класса аффинных нелинейных систем. Алгоритм основан на подходе *Model Predictive Path Integral* [3] и включает в себя идею нахождения параметров распределения семплирования управлений, при которых вероятность того, что элементы управления окажутся безопасными, превышает заранее определенный порог.

Результаты экспериментального тестирования подтверждают эффективность нашего метода. Важно отметить, что предлагаемый нами подход превосходит существующие современные методы предотвращения столкновений (*ORCA* [2] и *B-UAVC* [4]).

## 2. Предложенный подход

Предлагаемый подход основан на идеях алгоритма *Model Predictive Path Integral* (*MPPI*) [3]. Этот алгоритм был разработан для решения задачи поиска оптимального управления для нелинейных систем. Алгоритм *MPPI* берет схему методов *MPC*. Основное отличие от *MPC* заключается в том, что управление строится с использованием стохастического метода оптимизации. На каждом временном шаге *MPPI* создает набор управляющих последовательностей путем семплирования из нормального распределения с параметрами  $\mu, \Sigma$ . После чего получает набор соответствующих траекторий, используя заданную модель динамической системы. Каждая траектория оценивается с использованием некоторой функции оценки качества, и на основе полученных значений строится итоговое управляющее воздействие. Таким образом, алгоритм позволяет находить управления для нелинейных динамических систем и использовать нелинейные функции оценки качества [3].

Для поиска безопасных управлений предлагается добавить учет ограничений, которые будут гарантировать безопасность движения. Для нахождения таких ограничений был выбран алгоритм *Optimal Reciprocal Collision Avoidance* (*ORCA*) [1]. Он основан на идее нахождения линейных ограничений в области скоростей.

Далее, нам необходимо создать такую последовательность, которая бы, с одной стороны, продвигала бы агента в сторону целевой позиции, а с другой стороны, удовлетворяла найденным ограничениям безопасности. Рассмотрим подробнее предлагаемый метод учета линейных ограничений. Пусть дана дискретная модель движения агента в нелинейной аффинной форме:

$$(1) \quad \mathbf{x}_{t+1} = F(\mathbf{x}_t) + G(\mathbf{x}_t)\mathbf{u}_t$$

где  $\mathbf{x}_t \in \mathbb{R}^n$  – состояние агента в момент времени  $t$ ,  $\mathbf{u}_t = (u_{1,t}, u_{2,t}, \dots, u_{m,t}) \in \mathbb{R}^m$  управляющее воздействие в момент времени  $t$ ,  $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$  и  $G : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$

$\mathbb{R}^n$  функции, которые определяют модель движения агента. Компоненты вектора управления могут быть ограничены:  $u_{k,min} \leq u_{k,t} \leq u_{k,max}$ .

Для того, чтобы учесть линейные ограничения по предотвращению столкновений в процессе семплирования, необходимо найти новые параметры распределения  $\mu', \Sigma'$ , близкие к исходным  $\mu, \Sigma$ . При этом, новые параметры должны гарантировать, что вероятность выполнения линейного ограничения была выше определенного значения  $\alpha$  ( $\alpha \lesssim 1$ ). Эта задача может быть определена как задача оптимизации. Итак, целевую функцию задачи оптимизации можно сформулировать в следующем виде:

$$(2) \quad \begin{aligned} & \|\mu' - \mu\|_1 + \|\text{diag}(\Sigma') - \text{diag}(\Sigma)\|_1 \\ & \sigma_k'^2 \geq 0, \quad k = 1, \dots, m \end{aligned}$$

где  $\mu, \Sigma$  – заданные параметры распределения вектора независимых случайных величин  $u \sim \mathcal{N}(\mu, \Sigma)$ ;  $\mu', \Sigma'$  – искомые параметры распределения.  $\text{diag}(A)$  – это вектор, построенный из элементов главной диагонали некоторой матрицы  $A$  (т.е. для матрицы ковариации  $\Sigma'$  этот вектор будет содержать значения  $\text{diag}(\Sigma') = (\sigma_0'^2, \sigma_1'^2, \dots, \sigma_m'^2)$ ).

Далее необходимо учесть ограничения, полученные алгоритмом избегания столкновений *ORCA*. Пусть  $\mathcal{N}_i$  – это множество соседей некоторого агента  $i$  (т.е. множество агентов, наблюдаемых агентом  $i$ ). Зададим линейное ограничение, полученное агентом  $i$  для избегания столкновений с агентом  $j \in \mathcal{N}_i$  в форме неравенства. Таким образом, вектор скорости  $v^i = (v_x, v_y)$  удовлетворяет этому ограничению, если истинно следующее:

$$(3) \quad a_j \cdot v_x + b_j \cdot v_y + c_j \leq 0$$

С другой стороны,  $v^i = (v_x, v_y)$  для заданного состояния  $x_t^i$  и управления  $u_t^i$  может быть вычислен с использованием модели из уравнения (1). Поскольку состояние  $x_t^i$  включает позицию агента  $(p_x, p_y)$ , вектор скорости может быть переписан таким образом, чтобы линейно зависеть от управления  $u_t^i$ .

$$(4) \quad \begin{aligned} v_{x,t}^i &= F_1 + G_{1,1}u_{t,1} + \dots + G_{1,m}u_{t,m} - p_{x,t} \\ v_{y,t}^i &= F_2 + G_{2,1}u_{t,1} + \dots + G_{2,m}u_{t,m} - p_{y,t} \end{aligned}$$

Пусть  $\mathbf{a}'_j = \{(a_j G_{1,1} + b_j G_{2,1}), \dots, (a_j G_{1,m} + b_j G_{2,m})\}$  и  $b'_j = -(c_j + a_j(F_1 - p_{x,t}) + b_j(F_2 - p_{y,t}))$ . Тогда неравенство (3) может быть переписано в следующей форме:  $\mathbf{a}'_j{}^T \mathbf{u}_t \leq b'_j$ .

Семплируя управляющие последовательности, управление  $\mathbf{u}_t = \{u_{t,1}, \dots, u_{t,m}\}$  может рассматриваться как вектор с независимо нормально распределенными компонентами. Затем, используя теорию вероятностных ограничений (см. теорему 4.9 в разделе 4.3 из [5]), условие  $Pr\{\mathbf{a}'_j{}^T \mathbf{u}_t \leq b'_j\} \geq \alpha$  выполняется тогда и только тогда, когда  $\mathbf{a}'_j{}^T \mu' + \Phi^{-1}(\alpha) \sqrt{\mathbf{a}'_j{}^T \Sigma' \mathbf{a}'_j} \leq b'_j$ , где  $\mu', \Sigma'$  – математическое ожидание и ковариационная матрица случайного вектора  $\mathbf{u}_t$ ,  $\Phi(\cdot)$  – кумулятивная функция стандартного нормального распределения, а  $\alpha$  – требуемый доверительный уровень. Аналогично возможно учитывать ограничения на допустимые управления ( $u_{k,min} \leq u_{k,t} \leq u_{k,max}$ ):

Тогда, задача оптимизации для вычисления новых параметров распределения семплирования управляющих воздействий может быть записана в следующей форме:

$$\begin{aligned}
& \arg \min_{\mu', \Sigma'} \|\mu' - \mu\|_1 + \|\text{diag}(\Sigma') - \text{diag}(\Sigma)\|_1 \\
(5) \quad & \text{s.t. } \mathbf{a}_j'^T \mu' + \Phi^{-1}(\alpha) \sqrt{\mathbf{a}_j' \Sigma' \mathbf{a}_j'^T} \leq b'_j, \forall j \in \mathcal{N}'_i \\
& \sigma'_k \geq 0, k = 1, \dots, m
\end{aligned}$$

Таким образом, решая поставленную оптимизационную задачу, можно получить новые параметры для распределения семплированных управлений, так что с заданной вероятностью  $\alpha$  выбранные элементы управления будут удовлетворять ограничениям безопасности.

### 3. Экспериментальная оценка

Предложенный подход был реализован с использованием языка C++ и его эффективность была оценена в серии экспериментов в симуляции. В экспериментах использовалась следующая модель двухколесного робота с независимым приводом. Робот моделировался как круг радиуса  $r = 0.3$ . Подробнее модель движения робота описана в [2]. Сценарий эксперимента строился следующим образом. В начальный момент времени набор из  $n$  агентов расположен на равном расстоянии друг от друга в круге диаметром 12 м. Цель каждого агента – переместиться на позицию на противоположной стороне круга. Количество роботов в заданиях варьировалось от 2 до 15. Запуск задания считался успешным, если все агенты достигли целей без столкновений за заданный лимит в 1000 шагов симуляции. Запуск каждого задания с некоторым фиксированным количеством роботов повторялся 10 раз. В дополнение к предложенному методу (обозначим его как MPPI), в экспериментах так же участвовали алгоритмы предотвращения столкновений ORCA [2] и B-UAVC [4], адаптированные к рассматриваемой модели движения.

Основным показателем, измеренными в ходе эксперимента, было симуляционное время, за которое все агенты достигали своих целей (*makespan*). Результаты эксперимента приведены на рисунке 1.

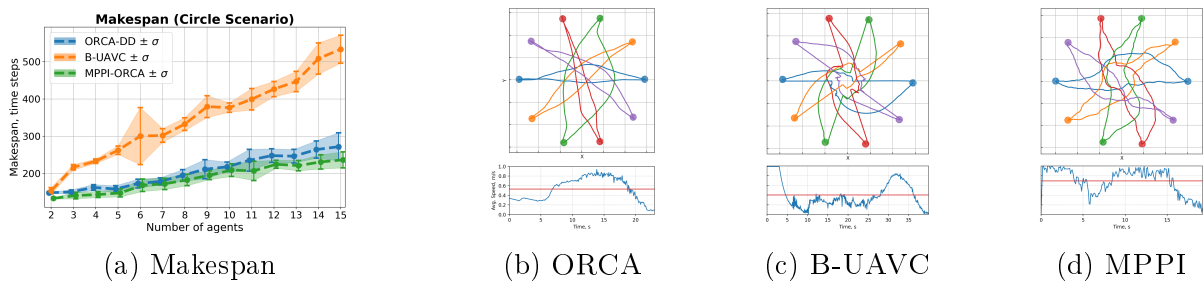


Рис. 1: (a) Среднее значение величины *makespan* и ее стандартное отклонение для сравниваемых алгоритмов в зависимости от числа роботов в сценарии. (b)–(d) Иллюстрация траекторий, полученных для задания с 10 роботами. Вверху показана визуализация траекторий. Ниже приведен график изменения средней скорости с течением времени (синяя линия) и средней скорости всех роботов за все время моделирования (красная линия)

В рамках эксперимента все алгоритмы успешно справились с поставленной задачей во всех случаях. Среднее значение *makespan* и стандартное отклонение для каждого числа агентов показаны на рисунке 1 (а). Можно видеть, предложенный алгоритм в среднем обеспечивает лучшие решения, чем два других подхода. И хотя результаты МРПИ и ORCA частично перекрывают друг друга, однако разрыв между ними растет с увеличением количества агентов. В то же время результаты алгоритма В-UAVC заметно хуже, чем у МРПИ и ORCA.

При детальном рассмотрении траекторий (иллюстрация траекторий в задании с 10 агентами приведена на рисунке 1 (b–d)) видно, что алгоритм ORCA находит незначительно более плавные траектории, но в некоторых случаях они имеют большую продолжительность, а средняя скорость движения ниже. В то же время МРПИ, благодаря своей стохастической природе, дает более флуктуирующие траектории, но продолжительность его решений меньше. Алгоритм В-UAVC генерирует колеблющиеся траектории с высокой стоимостью.

## 4. Заключение

В работе рассматривается задача децентрализованного избегания столкновений в много-агентных системах и предлагается метод решения, основанный на стохастическом подходе к поиску оптимальных управлений и учитывающий кинематические ограничения широкого класса динамических систем. В рамках работы было проведено экспериментальное сравнение предложенного подхода с существующими методами избегания столкновения. Результаты показали, что предложенный метод успешно справляется с задачей безопасного достижения целей группой агентов и позволяет находить более эффективные решения, по сравнению с другими методами

В будущей планируется включить учет неточности в информации о состоянии агента и его соседей, а так же неточности при выполнении управляющих воздействий. Кроме того, важным направлением дальнейших исследований является испытание предложенного метода в более продвинутых симуляторах и на реальных роботах.

Исследование выполнено за счет гранта Министерство науки и высшего образования Российской Федерации №075-15-2020-799.

## Список литературы

1. Van Den Berg J. et al. Reciprocal n-body collision avoidance // Robotics research. Springer, 2011. P. 3–19.
2. Snape J., et al. Smooth and collision-free navigation for multiple robots under differential-drive constraints // Proceedings of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2010). 2010. P. 4584–4589.
3. Williams G., et al. Aggressive driving with model predictive path integral control // 2016 IEEE International Conference on Robotics and Automation (ICRA). 2016. P. 1433–1440.
4. Zhu H., Alonso-Mora J. B-uavc: Buffered uncertainty-aware voronoi cells for probabilistic multi-robot collision avoidance // Proceedings of the International Symposium on Multi-Robot and Multi-Agent Systems (MRS2019). 2019. P. 162–168.
5. Liu B. Theory and practice of uncertain programming. Springer, 2009. Vol. 239.