

Децентрализованное управление строем

Хасан Хафизов

17 июня 2017 г.

Содержание

1	Введение	1
2	Постановка задачи	2
2.1	Описание	2
2.2	Формальная постановка	3
3	Алгоритм управления	3
3.1	Мастер	4
3.1.1	Оценка движения мастера по траектории	4
3.1.2	Выводы	6
3.2	Миньон	7
3.2.1	Воздействие на агента-лидера функцией Хевисайда	8
3.2.2	Выводы	8
4	Строй	11
4.1	Моделирование движения строя	13

1 Введение

С задачей совместного управления движением группы роботов чаще всего можно встретиться в военной сфере. Одним из перспективных направлений в этой области можно назвать управление группой БПЛА с целью разведки. Для этой задачи важен именно многоагентный подход, так как для систем противовоздушной обороны (ПВО) противника поражение большого количества целей является гораздо более сложной задачей, из-за возрастания необходимых вычислительных мощностей для одновременного сопровождения множество воздушных объектов. Вероятность выполнения задачи так же повышается, так как выход из строя

части роботов не окажет критического влияния на группу в целом. Дополнительный выигрыш от строевого движения можно получить задав конфигурацию строя так, чтобы за впереди идущими агентами образовывалась зона пониженного давления, давая тем самым возможность остальным участникам группы экономить топливо или заряд батарей.

2 Постановка задачи

2.1 Описание

В этой работе будет решаться задача управления группой роботов, объединённых в строй с произвольным порядком (рисунком). Каждый робот в строю является интеллектуальным агентом, движущимся в соответствии со своим законом управления, который направлен, во-первых на то, чтобы выполнить общую задачу, а во-вторых чтобы сохранить исходную структуру строя. В качестве задачи, которая будет выполняться строем была выбрана задача движения по произвольной траектории с задаваемой скоростью.

Для взаимозаменяемости требования с аппаратной точки зрения ко всем агентам одинаковые, однако программно в каждый момент времени агент будет принадлежать одному из двух классов:

- Мастер — агент, задача которого отрабатывать движение по траектории и быть главным ориентиром для всех остальных агентов.
- Миньон — агент, задача которого сохранять своё местоположение относительно других участников группы, тем самым сохраняя структуру строя

Моделирование в работе будет проводиться в горизонтальной 2D плоскости. Моделями агентов будут материальные точки с массой.

Ограничения:

- У каждого агента есть ограничение видимости. Если два участника строя в данный момент времени находятся вне зоны видимости друг-друга, то они не могут напрямую обмениваться информацией
- Максимальное по модулю управляющее воздействие ограничено
- Агенты передают друг-другу информацию только о своих абсолютных координатах

Основные требования к решению:

- Без возмущений строй должен отрабатывать заданную траекторию с малыми погрешностями
- При малых возмущениях каждый отдельно взятый агент не должен существенно отклоняться от своей позиции в строю. Строй в целом должен по-прежнему выполнять задачу, с погрешностями соразмерными возмущениям
- При краткосрочных больших возмущениях, даже если строй потерял целостность, он должен самовосстанавливаться и продолжать выполнение задачи
- При выходе из строя одного или нескольких агентов строй должен переформироваться так, чтобы сохранить исходный рисунок

2.2 Формальная постановка

Вектор состояния каждого агента в горизонтальной плоскости xy :

$$X = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix},$$

где $x_1 = x$, $x_2 = \dot{x}$, $x_3 = y$, $x_4 = \dot{y}$. Тогда закон движения:

$$\dot{X} = AX + BU$$

3 Алгоритм управления

В предлагаемом мной алгоритме управления агентов можно разделить на два класса:

- мастер — агент, который отрабатывает движение по траектории
- миньон — поддерживает своё местоположение в строю

Закон управления для этих двух типов агентов задаётся по-разному.

3.1 Мастер

Мастером является агент, для которого желаемый закон движения S_d задаётся оператором извне: это может быть записанная в память агента траектория, целевая позиция или скорость.

Фактически, этот агент ничего не знает о существовании других агентов в строю (миньонов). Его задача — выполнение поставленного закона движения, поэтому закон управления зависит только от закона движения:

$$\vec{u} = \vec{u}(S_d)$$

Рассмотрим конкретный закон управления \vec{u}_{tr} : движение по некоторой траектории $\vec{tr}(t)$:

$$\vec{u}_{tr} = \vec{u}_{along} + \vec{u}_{across}$$

Закон управления состоит из двух частей. Первая \vec{u}_{along} отвечает за усилие вдоль траектории, вторая \vec{u}_{across} — поперёк. Направлением для \vec{u}_{along} служит направление вектора между текущим положением агента и следующей точкой траектории.

(Здесь будет более подробно о том, как вычисляется следующая точка траектории, о алгоритмах управления на PD регуляторах, которые используются как в \vec{u}_{along} , так и в \vec{u}_{across})

Пример движения мастера по траектории, задаваемой параметрическим уравнением, где s — параметр: $s \in [0, 5000]$.

$$x(s) = 300 \cdot \cos\left(\frac{s}{300}\right); \quad y(s) = s$$

3.1.1 Оценка движения мастера по траектории

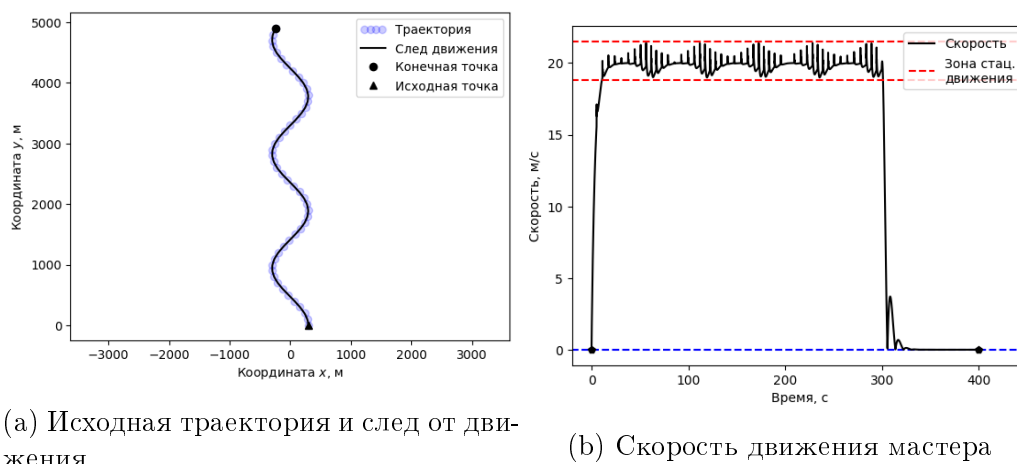
Агент преодолел заданную траекторию за $t = 331\text{с}$.

Проеденное расстояние:

$$S = \int_0^{5000} \sqrt{x'^2(s) + y'^2(s)} \, ds \approx 6051\text{м}$$

Средняя скорость $v_{av} = 18.3 \frac{\text{м}}{\text{с}}$.

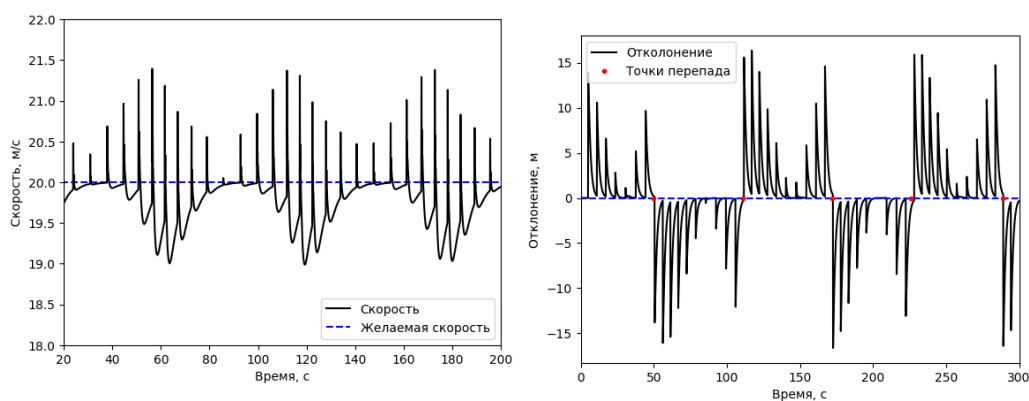
Стационарным режимом движения можно назвать режим, при котором скорость агента колеблется в пределах между $18.8 \frac{\text{м}}{\text{с}}$ и $21.5 \frac{\text{м}}{\text{с}}$. Более подробно скорость мастера в стационарном режиме можно увидеть на рис. 2а



(а) Исходная траектория и след от движения

(b) Скорость движения мастера

Рис. 1: Движение мастера по заданной траектории с заданной скоростью $v_{desired} = 20 \frac{м}{с}$. Расстояния на рисунках задаются в метрах, время в секундах, скорость в $\frac{м}{с}$



(а) Скорость мастера в стационарном режиме

(b) Отклонения мастера от траектории

Рис. 2: Иллюстрация отклонений от желаемого закона движения

Отклонения агента от заданной траектории представлены на рис. 2b. Выше нуля — отклонения от траектории вправо, ниже нуля — влево. Максимальное отклонение составляет примерно 15 м.

Резкие перепады на графике, обозначенные как точки перепада объясняются тем, что в этих точках у траектории изменяется знак первой производной, а агент, всегда остающийся на внутренней части траектории резко оказывается на внешней. Проверим это утверждение. Заданное выше параметрическое уравнение фактически является уравнением

$$x = 300 \cdot \cos\left(\frac{y}{300}\right)$$

Равенство нулю первой производной:

$$\sin\left(\frac{y}{300}\right) = 0 \Rightarrow y = 300\pi n$$

Найдём первые 5 точек траектории, в которой происходит смена знака второй производной:

$$M = \left\{(-300; 942), (300; 1885), (-300; 2827), (300; 3770), (-300; 4712)\right\}$$

Найдём моменты времени, в которые эти точки будут достигнуты агентом:

$$\hat{L}_t = \{61c, 118c, 177c, 234c, 292c\}$$

Реальные же моменты времени, в которые просходит резкое изменение величины отклонения от траектории:

$$L_t = \{50c, 110c, 172c, 226c, 289c\}$$

То есть, изменение стороны относительно линии траектории по которой движется агент изменяется незадолго до того, как будет изменён знак первой производной траектории. Эта закономерность так же наблюдается на рис. 3.

(Тут будет исследование устойчивости алгоритма к случайным возмущениям)

3.1.2 Выводы

Движение мастера по траектории является точным и предсказуемым. Данный алгоритм управления по траектории с задаваемой скоростью является пригодным для применения.

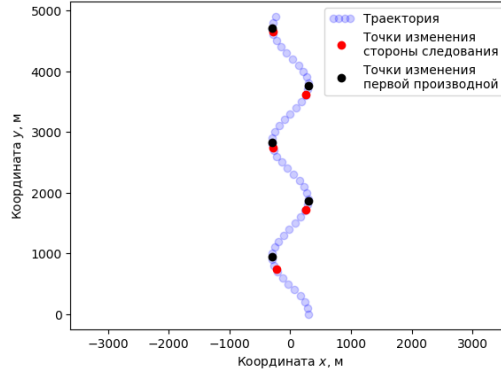


Рис. 3: Траектория с наложенными на неё точками изменения стороны следования и точками изменения знака первой производной.

3.2 Миньон

Миньон является ведомым агентом, во время движения он никак не использует информацию о траектории движения. Всё, что знает миньон i — это положение своего агента-лидера L . Относительно агента-лидера миньон выстраивает свой закон управления u_m :

$$u_m = u_m(L)$$

Более подробно об агентах-лидерах будет сказано в гл. 4, а пока о них можно думать просто как о постоянно обновляемой точке, относительно которой агент строит некую другую точку, в которую стремится попасть. Эта точка будет называться виртуальным лидером L^v . Точка виртуального лидера своя у каждого миньона и зависит только от реального положения агента-лидера L .

Пусть, есть миньон с координатами \vec{A}_1 и виртуальным лидером в точке \vec{L}_1^v . Первый закон управления, который приходит на ум может выглядеть следующим образом:

$$u_m(t) = S \cdot P, \text{ где } S = (\vec{L}_1^v - \vec{A}_1) \quad (1)$$

Суть такого закона управления в том, что воздействие осуществляется в направлении виртуального лидера с усилием пропорциональным расстоянию между текущей координатой и координатой виртуального лидера. Коэффициент P — пропорциональная составляющая PID регулятора.

Однако, такой закон управления не выдерживает некоторых испытаний.

3.2.1 Воздействие на агента-лидера функцией Хевисайда

Этот численный эксперимент заключается в следующем: есть пара агентов — миньон и его агент-лидер. На агента-лидера воздействует сила по закону Хевисайда, то есть, фактически, после некоторого момента времени \hat{t} появляется постоянное воздействие на агента-лидера.

Для облегчения понимания ситуации рассмотрим воздействие на примере.

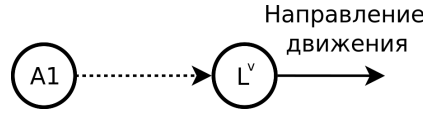


Рис. 4

На рис. 4 изображён миньон A_1 и его виртуальный лидер L^v . На L^v действует сила (распределённая как функция Хевисайда):

$$F(t) = \begin{cases} 1, & \text{если } t \geq 5\text{с} \\ 0, & \text{в остальных случаях} \end{cases}$$

Если использовать закон управления, записанный уравнением (1), то возникнет ситуация, при которой расстояние между миньоном и его виртуальным агентом будет постоянно расти. Ускорение лидера будет расти быстрее, чем ускорение миньона из-за того, что агент строит свой закон управления только на основании положения лидера, ничего не зная о его ускорении и скорости. Однако передавать эти данные от лидера к миньону может оказаться слишком накладно, т.к. придётся поддерживать более широкий канал связи.

В качестве альтернативы миньон может использовать уже имеющиеся данные о положениях виртуального лидера и увеличивать управляющее воздействие, если расстояние между ним и лидером растёт \dot{S} . Так же для устранения статических ошибок добавляется I составляющая PID регулятора.

$$\vec{u}_m(t) = P \cdot S + I \cdot \int_0^t S dt + D \cdot \dot{S} \quad (2)$$

Результаты численного моделирования с данным законом управления:

3.2.2 Выводы

Описанный закон управления справляется с задачей приведения миньона в точку виртуального лидера и будет использован в дальнейшем для управления строем.

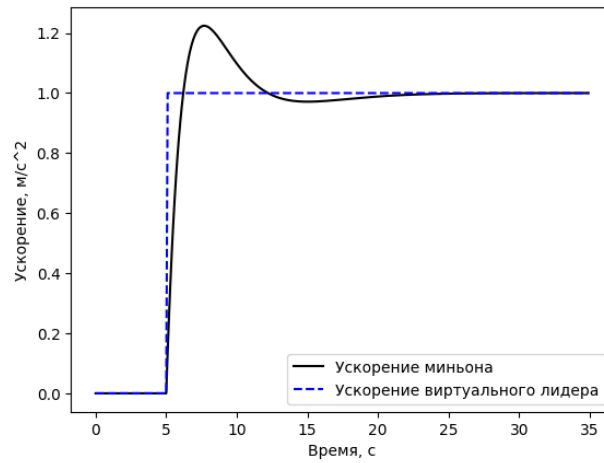


Рис. 5: Ускорение миньона и его виртуального лидера. Виртальный лидер ускоряется по закону Хевисайда. Ускорение выходит на стационарный режим на 25с: через 20с. после воздействия.

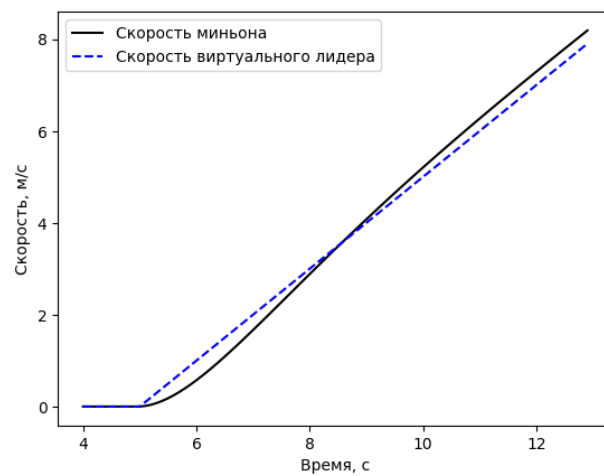


Рис. 6: Скорость миньона и виртуального лидера

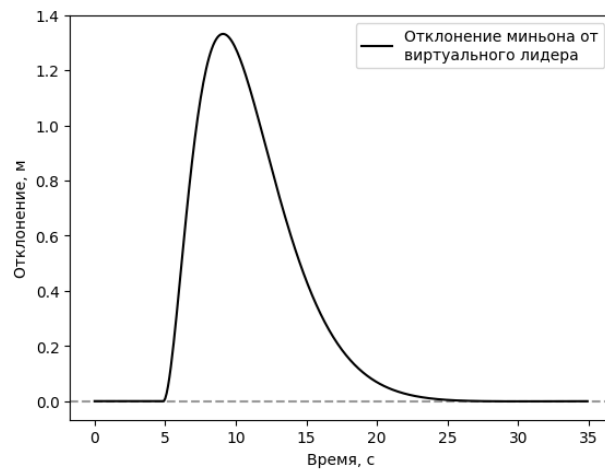


Рис. 7: Отклонение положения миньона от его виртуального агента

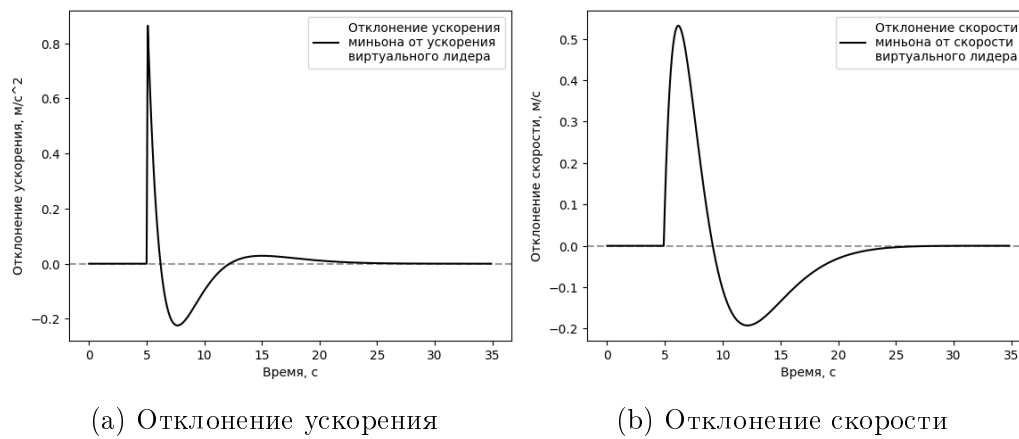


Рис. 8: Иллюстрация отклонений от закона движения виртуального лидера. Выше нуля — значение меньше, чем у лидера, ниже нуля — выше, чем у лидера.

4 Строй

Строй представляет из себя множество агентов соединённых связями, подобно графу. Для каждого агента, не являющегося мастером должно быть определено не менее одного агента-лидера. Лидером для агента может быть как мастер, так и любой миньон. Каждый агент — вершина графа, каждая связь миньон-лидер — ребро графа. Пример строя из 5-ти агентов изображён на рис.9.

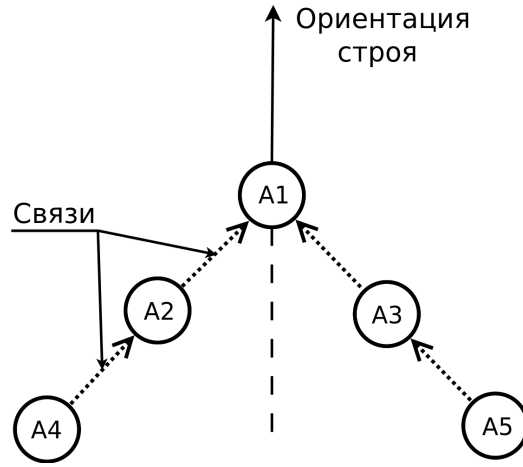


Рис. 9: Пример клиновидного строя из 5-ти агентов

Каждая связь J_{ij} — это вектор в полярных координатах, у которого радиус — это расстояние от миньона i до миньона лидера j , а угол — это угол между изначальной ориентацией строя и вектором, соединяющим миньона с его лидером. К примеру, если расстояния между каждой парой агентов равно 1м, то связи:

- J_{1j} — не определены ни для каких j : A_1 является мастером
- $J_{21} = (r = 1\text{м}; \varphi = 225^\circ)$ — для миньона A_2 агентом лидером является мастер A_1 . Агенты находятся на расстоянии 1м, угол между соединяющим их вектором и вектором изначальной ориентации составляет 225°
- J_{31}, J_{42}, J_{53} — определяются аналогично

При повороте строя поворачивается и вектор ориентации строя, за счёт чего далее пересчитывается позиция, в которой должен находиться агент после поворота. Эта позиция будет именоваться виртуальным лидером.

Виртуальный лидер — точка пространства, в которую стремиться попасть миньон для того, чтобы поддерживать фигуру строя.

Например, на рис. 10 изображён поворот строя на угол b . Для того, чтобы пересчитать позицию, в которой должен находиться миньон после поворота (виртуальный лидер) необходимо повернуть вектор изначальной ориентации O_0 на угол $b + \varphi$, где φ — угол из любого вектора связи J_{ij} миньона i , далее от положения выбранного миньона-лидера необходимо отложить расстояние равное r из вектора связи в направлении повернутого вектора изначальной ориентации O_0 .

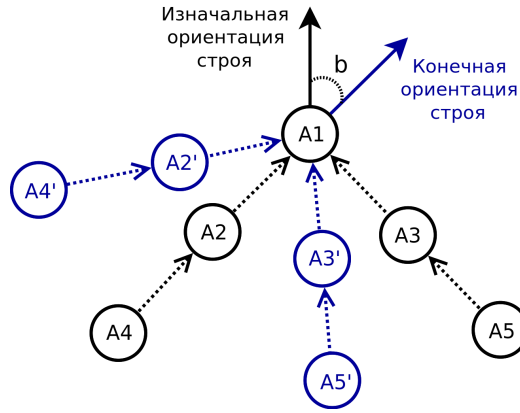


Рис. 10: Поворот строя на угол b

Рассмотрим пример того, как будет рассчитан виртуальный лидер A'_5 для агента A_5 (см. рис. 10), для которого определён вектор связи $J_{53} = (r_{53}; \varphi_{53})$ (вектор связи с A_3). Для начала повернём вектор изначальной ориентации O_0 :

$$\vec{O}_1 = \begin{pmatrix} \cos(b + \varphi_{53}) & \sin(b + \varphi_{53}) \\ \sin(b + \varphi_{53}) & \cos(b + \varphi_{53}) \end{pmatrix} \vec{O}_0$$

Теперь вдоль полученного вектора \vec{O}_1 необходимо отложить от позиции агента A_3 расстояние равное r_{53} :

$$\vec{A}'_5 = \vec{A}_3 + \frac{\vec{O}_1}{|\vec{O}_1|} r_{53}$$

В приведённых выше расчётах неявно предполагалось, что агент A_3 на момент пересчёта виртуального лидера для A_5 уже находится в точке A'_3 . До тех пор, пока не будет выполнено это условие получаемое A'_5 не будет совпадать с тем, что приведено на рисунке. Однако, это ожидаемая

ситуация, т.к. A_5 не знает ни о каких агентах кроме A_3 , то он и полагается только на данные, получаемые от A_3 , а если A_3 ещё не достиг точки виртуального лидера, то и в координатах виртуального лидера A'_5 будут содержаться ошибки.

Описанную выше проблему несложно решить, если строить виртуальных лидеров на основе позиции не агента-лидера, а на основе точки виртуального лидера агента-лидера. Однако, после поворота у миньонов находящихся дальше от центра поворота (центр поворота — мастер) расстояние до их виртуальных лидеров окажется больше, чем у миньонов ближе к центру поворота (см. $A_3 - A'_3$ в сравнении с $A_5 - A'_5$ на рис. 10), что приведёт к тому, что пропорциональная составляющая PID регулятора вырабатывающего управление для следования миньонов в точку виртуального лидера выдаст команду создать большее управляющее воздействие, следовательно АКБ агентов будут разряжаться более неравномерно, чем при более инертном расчёте виртуальных лидеров от позиции агентов-лидеров.

4.1 Моделирование движения строя

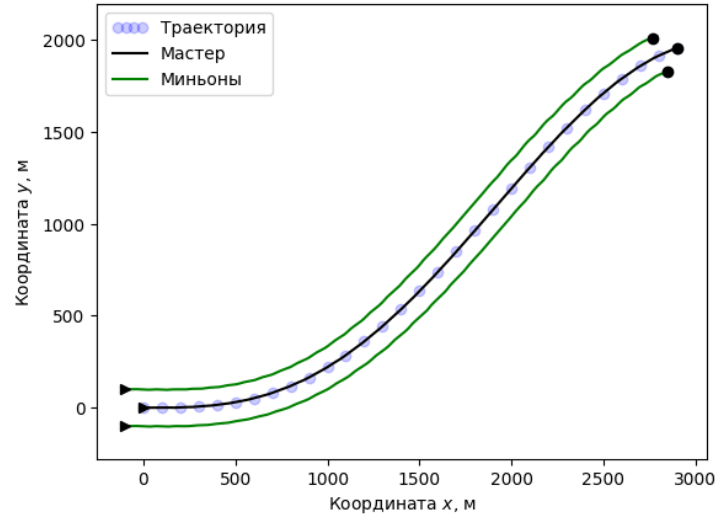


Рис. 11: Движение строем мастера и двух миньонов