

УДК 519.2

Р. А. Гиргидов¹

ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ РОЯ С ИСПОЛЬЗОВАНИЕМ ПРОИЗВЕДЕНИЯ АДАМАРА

Использование большого количества дронов ставит перед нами задачи корректного, прозрачного и высокопроизводительного моделирования их совместной деятельности в рамках роевого поведения. Для этого была разработана и приведен пример перехода от описания системы управления единичного дрона к поведению роя одинаковых дронов на основе алгебры (произведения) Адамара. Было выяснено, что использование подобного подхода дает повышение производительности модели в десятки раз и позволяет производить моделирование роевых систем с очень большим количеством особей.

Ключевые слова: произведение Адамара, многоагентная система.

DOI: 10.55959/MSU/0137-0782-15-2023-47-3-10-22

1. Введение. Использование дронов в современной жизни приобретает все больший масштаб и в целом перешло из разряда исследований в прикладную область, но исследования по автономным дронам и, в частности, по дронам, объединенным в рой, не только активны, но со временем приобретают все большую значимость. Происходит это из-за того, что при использовании большого количества дронов на передний план начинают выходить не столько проблемы телеуправления и компоновки отдельных дронов, сколько проблемы взаимодействия дронов друг с другом, особенно осуществляемые в автономном режиме. Таким образом, взаимодействие дронов внутри роя является важным частным случаем автономного взаимодействия роботизированных систем, при этом важно еще и исследование предполагаемых синергетических эффектов, связанных с количеством и качеством взаимодействий между ними [1].

Традиционно, при описании поведения и моделировании роя используют многоагентную имитационную модель. Для более полной классификации приведем здесь некоторые определения.

Имитационной моделью называют логико-математическое описание объекта, которое может быть использовано для экспериментирования на компьютере в целях проектирования, анализа и оценки функционирования объекта.

Как правило, к имитационным моделям относят модели сложных систем, где известно поведение системы или частей системы в определенных условиях, но целиком описание системы невозможно или очень сложно. Объединение таких частных моделей зачастую выявляет неизвестные ранее связи и особенности поведения моделируемого объекта.

Однако необходимо отметить, что термин “имитационное моделирование” практически не используется в англоязычной литературе [2].

В России к имитационным моделям относят: дискретно событийные (например, модели системы массового обслуживания), модели системной динамики (как правило, имеющие явную схему решения и пригодные для визуализации в реальном времени) и агентные (исследующие взаимодействие похожих агентов).

Моделирование роя обычно относят к имитационному моделированию, так как в основе его лежит многоагентное моделирование. Исследования роевого поведения групп дронов заключаются в выявлении глобальных особенностей роя на основе правил и законов, действующих на каждый дрон в отдельности. Однако многоагентный подход имеет ряд серьезных недостатков, связанных, в первую очередь, с эмпиричностью модели и невозможностью теоретически предсказать свойства роя для получения желаемого поведения. Целью данной работы является применение парадигмы моделирования роя не как набора отдельных агентов, а как единого объекта

¹ Санкт-Петербургский политехнический университет Петра Великого, ст. преп., e-mail: ruben@flightpath3d.com.ru

исследования. В частности, предлагается набор формальных процедур, позволяющих перейти от существующих многоагентных роевых архитектур к имитационным моделям системной динамики. Данные процедуры должны обеспечить отображение поведения одного дрона в рое на модели роя, обеспечивая повышение производительности модели роя в целом.

Хотя многоагентное моделирование дает вполне разумные результаты, такой способ моделирования не достаточно эффективен и не дает возможности лаконично описать работу роя в целом, а также выявить закономерности его поведения. Нами предлагается методика преобразования описания поведения одного дрона в модель роя целиком на базе произведения Адамара.

Мы исходили из того, что для описания системной динамики роя, необходимо использовать матричное описание модели роя. Однако, в данном случае, стандартные операции линейной алгебры над матрицами не позволили нам получить простое и “прозрачное” описание модели роя, а также разработать формальную процедуру перехода от модели одного дрона и его взаимодействия с соседями к общему описанию роя.

Для решения указанной проблемы, предлагается использовать матричные операции, основанные на произведении Адамара, дополнив его операциями над матрицами различных размерностей, вводя некоторые неканонические, но часто применяемые на практике операции над матрицами.

В рамках данной статьи мы не ставим себе целью провести полное исследование алгебры, основанной на произведении Адамара, а воспользуемся только некоторыми прикладными результатами.

2. Основные понятия и обозначения.

2.1. Алгебраические операции. Рассмотрим основные положения произведения Адамара. Пусть матрицы A и B :

$$A = [a_{ij}]_{mn} = \begin{bmatrix} a_{11} & \dots & a_{1m} \\ \vdots & \ddots & \vdots \\ a_{n1} & \dots & a_{nm} \end{bmatrix}, B = [b_{ij}]_{mn} = \begin{bmatrix} b_{11} & \dots & b_{1m} \\ \vdots & \ddots & \vdots \\ b_{n1} & \dots & b_{nm} \end{bmatrix} \quad (1)$$

— прямоугольные матрицы одинаковой размерности $(m \times n)$. Тогда введем следующие произведения.

Определение 1. Произведением Адамара будем называть поэлементное умножение матриц $A \circ B = a_{ij}b_{ij}$.

Определение 2. Единичной матрицей будем считать матрицу $E = (1)_{ij}$, все элементы которой равны единице.

Расширим данное определение, определяя его для произведения прямоугольной матрицы на матрицу столбец и матрицу строку.

Определение 3. Введем понятие произведения Адамара для столбца. Пусть $X = [x_1, \dots, x_n]^T$ — столбец размерности n . Будем говорить, что

$$A \circ X = a_{ij}x_i = \begin{bmatrix} a_{11} & \dots & a_{1m} \\ \vdots & \ddots & \vdots \\ a_{n1} & \dots & a_{nm} \end{bmatrix} \circ \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} a_{11}x_1 & \dots & a_{1m}x_1 \\ \vdots & \ddots & \vdots \\ a_{n1}x_n & \dots & a_{nm}x_n \end{bmatrix}, \quad (2)$$

т.е. каждый столбец матрицы A умножается поэлементно на столбец матрицы X . Для данного определения произведения матрицы и столбца необходимым условием является равная размерность столбцов матрицы и столбца, на который умножается матрица.

Определение 4. Произведением Адамара для строки будем называть следующую операцию. По аналогии со столбцом определим умножение на строку. Пусть $Y = [y_1 \dots y_m]$ — строка размерности m , тогда

$$Y \circ A = a_{ij}y_j = [y_1 \dots y_m] \circ \begin{bmatrix} a_{11} & \dots & a_{1m} \\ \vdots & \ddots & \vdots \\ a_{n1} & \dots & a_{nm} \end{bmatrix} = \begin{bmatrix} a_{11}y_1 & \dots & a_{1m}y_m \\ \vdots & \ddots & \vdots \\ a_{n1}y_1 & \dots & a_{nm}y_m \end{bmatrix}. \quad (3)$$

Таким образом, при подобных умножениях размерность результирующей матрицы будет совпадать с размерностью исходной матрицы, умноженной на вектор (записанный в виде строки или столбца), и равняться $(m \times n)$.

2.2. Функциональные и дифференциальные операции на основе алгебры Адамара. Функция, действующая на матрицу в рамках данной алгебры, может быть представлена в виде $f(A)_{m \times n} = [f(a_{ij})]_{m \times n}$, т.е. функция действует на каждый элемент матрицы. Символом F будем обозначать гиперфункцию $F(A) = [f_{ij}(a_{ij})]$, т.е. на каждый элемент матрицы a_{ij} действует своя функция f_{ij} . Таким образом, обычная функция, действующая сразу на всю матрицу, будет частным случаем гиперфункции, где все $f_{ij}(a_{ij}) = f(a_{ij})$, т.е. одинаковы. Также мы будем пользоваться определением дифференциала функции из [3]. На основании этого определения производная по времени будет рассчитываться следующим образом:

$$\frac{dF}{dt} = \left[\frac{df_{ij}}{dt} \right].$$

Дифференциал матрицы будем задавать как $dF = [df_{ij}]$.

Определение 5. По аналогии с определением функции введем определение обратной матрицы и опишем обратную матрицу как функцию возведения в степень: $A^{-1} = (a^{-1})_{ij}$.

Как видно из формулы, матрица A не должна иметь нулевых элементов, однако это не очень удобно при практическом использовании, поэтому нами предлагается несколько изменить данное определение, расширив его на матрицы с нулевыми элементами:

$$A^{-1} = \begin{cases} a_{ij}^{-1}, & a_{ij} \neq 0, \\ 0, & a_{ij} = 0. \end{cases} \quad (4)$$

Определение 6. Назовем данную операцию расширенным матричным делением, а матрицу, полученную таким образом, “псевдообратной”.

Все приведенные операции справедливы для любых матриц размерностью $m \times n$. Умножение на константу будем рассматривать как частный случай функции, действующей на матрицу.

На основании данных выше определений можно сделать следующие выводы: адамарово произведение матриц одинаковой размерности и матрицы, умноженной на столбец или строку соответствующих размерностей, коммутативно и дистрибутивно;

$$A_{nm} \circ X_n + B_{nm} \circ X_n = (A_{nm} + B_{nm}) \circ X_n; \quad (5)$$

$$A \circ (X_n^{(1)} + X_n^{(2)}) = A \circ X_n^{(1)} + A \circ X_n^{(2)}. \quad (6)$$

3. Специальные матричные операторы. Введем также оператор понижения размерности матриц. Данную операцию часто называют сверткой.

Определение 7. Определим свертку по столбцу как

$$\langle A \rangle_- = \sum_{i=1}^n a_{ij} = \left[\sum_{i=1}^n a_{i1} \quad \dots \quad \sum_{i=1}^n a_{im} \right], \quad (7)$$

где горизонтальная линия в индексе указывает, что результат будет в виде строки, т.е. $\langle A \rangle_-$ — матрица-строка, где каждый ее элемент равен сумме элементов, соответствующего столбца матрицы.

Определение 8. Аналогично введем свертку по строкам как

$$\langle A \rangle_| = \sum_{j=1}^m a_{ij} = \left[\begin{array}{c} \sum_{j=1}^m a_{1j} \\ \vdots \\ \sum_{j=1}^m a_{nj} \end{array} \right], \quad (8)$$

где вертикальная линия в индексе указывает, что результат будет в виде столбца, т.е. $\langle A \rangle_{\downarrow}$ — матрица-столбец, где каждый ее элемент равен сумме элементов соответствующей строки матрицы A .

Данные соотношения выражаются через стандартное матричное умножение [4] следующим образом: $\langle A \rangle_{\downarrow} = E_{\downarrow} \cdot A$, $E_{\downarrow} = [1, 1, \dots, 1]$, $\langle A \rangle_{\downarrow} = A \cdot E_{\downarrow}$, $E_{\downarrow} = [1, 1, \dots, 1]^T$, где $E_{\downarrow}$ — матрица-столбец, $E_{\downarrow}$ — матрица-строка.

Для корректного описания роевого поведения в случаях различных характеристик дронов в рамках одного роя, описания правил переключения системы управления или логических условий взаимодействия дронов необходимо ввести матричный аналог индикаторной функции.

Для этого зададим гиперфункцию

$$\mathbb{1}_L(a) = \begin{cases} 1, & a \in L, \\ 0, & a \notin L, \end{cases}$$

где $L \subseteq \Phi$ — выбранное подмножество произвольного множества Φ . В матричном виде данная функция будет представлена как

$$\mathbb{1}_L(A) = \begin{bmatrix} 1, & a_{ij} \in L_{ij} \\ 0, & a_{ij} \notin L_{ij} \end{bmatrix}. \quad (9)$$

Для дальнейшего описания методики перехода от правил взаимодействия двух дронов к описанию роя на основании введенных определений необходимо будет рассмотреть модель одного дрона и его взаимодействия с другими дронами.

4. Описание модели дрона. Далее в работе рассматривается quadro- или октокоптер с максимальной скоростью полета около 12 м/с. Ограничение скорости в рассматриваемой ниже модели осуществляется двумя способами: прямое ограничение (на практике модель никогда не доходила до срабатывания этого ограничения) и косвенное ограничение (искусственно введенная виртуальная сила вязкого трения). Принципиальной особенностью данного вида дрона (и соответственно модели) является его осесимметричность.

Дрон оборудован оптическими датчиками, позволяющими на близком расстоянии (до r_a) определять скорость и координаты дрона соседа или соседей. Их координаты в данной области определяются с некоторой погрешностью (до ± 5 см), с частотой, достаточной для определения относительной скорости каждого дрона-соседа. Это зона уверенной детекции. На больших расстояниях (до r_l) мы можем определить только наличие (отсутствие) дронов в секторе. Конструктивно детекция в секторе представляется фотоэлементом или ультразвуковым датчиком относительно узкоспектрального излучения, направленного в одну сторону и определяющего наличие и интенсивность сигнала от другого дрона. Каждый дрон излучает сигнал в соответствующем спектре. Причем, если дронов в области видимости датчика несколько, то сигнал от них “суммируется” в датчике. Таким образом, получается, что излучение от нескольких далеко стоящих дронов датчиком будет восприниматься как излучение одного, но находящегося ближе и, соответственно, более яркого излучателя. В приведенном ниже примере дрон будет иметь 4 датчика, направленных в разные стороны с растровым углом обзора 90° , модель будет иметь 4 непересекающихся круговых сектора.

Предполагается, что дроны имеют автопилот, благодаря которому все дроны роя находятся на одной высоте. Автопилот присутствует, но может только стабилизировать дрон в горизонтальном положении в отсутствие воздействий от различных управляющих сил и зафиксировать высоту полета дрона. Обратим особое внимание на то, что стабилизация дрона не предполагает остановки дрона, так как у автопилота нет информации о начальной скорости и положении дрона в пространстве и всегда присутствует “паразитный” дрейф.

Движение дрона как материальной точки будет происходить на основании закона Ньютона в дифференциальной форме под действием квазифизических управляющих сил. В данном примере мы будем рассматривать плоскую задачу в предположении, что всем автопилотам роя дали команду держать определенную высоту. В этом случае движение дрона определяется уравнением

$$m \frac{d^2 \vec{r}}{dt^2} = \vec{F}_{fr} + \vec{U}_{ctr} + \vec{F}_{rel} + \vec{F}_{rnd},$$

где $\vec{r} = \begin{pmatrix} x \\ y \end{pmatrix}$ — вектор координат дрона в абсолютной системе отсчета, m — масса дрона, F_{fr} — результирующая сила взаимодействия одного дрона с другим. Данная сила, как и сила трения, относится к внутренним управляющим воздействиям и зависит от взаимного расположения дронов. Пусть U_{ctr} — вектор сил, получаемых от системы управления при обработке внешнего управляющего (целеполагающего) сигнала, ответственного за выполнение задачи (миссии), в отличие от внутренних управляющих сигналов, ответственных за взаимную навигацию дронов. Таким образом, все остальные силы и управляющие команды являются результатом срабатывания правил системы управления при изменении состояния окружающей среды или состояния дрона. Пусть F_{rnd} — случайное воздействие на дрон, оно может быть как искусственно созданной силой с помощью управляющего модуля, так и силой естественного происхождения, например от порывистого ветрового воздействия или “паразитных” сил, возникающих из-за особенностей работы (неточности) датчиков, а также запаздывания сигнала. Введение данной составляющей управляющих сигналов позволяет добиться получения наиболее устойчивой формации роя, толерантной к большинству внешних воздействий. В данном случае мы используем белое шумное воздействие.

Начальные условия задаются как координаты и скорость дрона в начальный момент времени:

$$\vec{r}_0 = \begin{pmatrix} x_0 \\ y_0 \end{pmatrix}, \vec{v}_0 = \begin{pmatrix} v_{x0} \\ v_{y0} \end{pmatrix}.$$

Рассмотрим силы, действующие на дрон, более подробно. Равнодействующая F_{fr} от управляющих сил, ответственных за гашение колебаний, моделируется по аналогии с силой вязкого трения и в этом случае будет иметь вид

$$\vec{F}_{fr} = -\mu_{ctr} \vec{V}_{fr},$$

где $\vec{V}_{fr}$ мы рассматриваем как равнодействующий вектор скоростей относительно дрона или дронов, находящихся в зоне уверенной детекции, μ_{ctr} — управляющий “коэффициент трения”, который подбирается экспериментально и зависит от требований, предъявляемых системе управления, связанной с взаимной навигацией дронов.

В случае множества дронов, влияющих на текущий дрон, силу взаимодействия между дронами можно представить как сумму сил, создаваемых всеми дронами, находящимися в зоне детекции:

$$\vec{F}_{rel} = \sum_i^{N(t)} \vec{F}_{rel_i}.$$

Здесь $N(t)$ — число дронов, находящихся в зоне детекции исследуемого объекта в каждый момент времени t . Если рассмотреть описание взаимодействия двух дронов, в данном случае влияния одного дрона на рассматриваемый, то $N = 1$.

Как было указано выше, все силы, воздействующие на дрон, являются квазифизическими, будучи по сути управляющими воздействиями, но эти воздействия в рамках представленной модели копируют физический процесс — взаимодействие атомов в процессе синтеза кристалла (формирования кристаллической решетки). Это позволяет надеяться на выстраивание дронов в рое в определенную структуру (определяемую параметрами указанных сил).

Отметим важную особенность работы дрона, основанную на физических ограничениях устройства: результирующая сила не может быть больше некой максимальной величины F_{max} , определяемой индивидуально для каждой модели дрона. В этом случае величина силы становится постоянной и равной максимуму, а вот направление воздействия должно совпадать с направлением результирующей всех управляющих сил, действующих на дрон:

$$\vec{F}_{res}^{\times} = \begin{cases} \vec{F}_{res}, & \left| \vec{F}_{res} \right| \leq F_{max}, \\ F_{max} \frac{\vec{F}_{res}}{\left| \vec{F}_{res} \right|}, & \left| \vec{F}_{res} \right| > F_{max}. \end{cases} \quad (10)$$

По аналогии с максимальной силой в процессе решения дифференциального уравнения вводится максимально возможная скорость:

$$\vec{V}_{res}^{\times} = \begin{cases} \vec{V}_{res}, & \left| \vec{V}_{res} \right| \leq V_{\max}, \\ V_{\max} \frac{\vec{V}_{res}}{|\vec{V}_{res}|}, & \left| \vec{V}_{res} \right| > V_{\max}, \end{cases} \quad (11)$$

но в рамках нашей модели воздействие силы вязкого трения таково, что введение ограничения по скорости практически не используется.

Данные особенности модели дрона описаны в работе [5].

Таким образом, в абсолютных координатах уравнение будет выглядеть так:

$$\begin{aligned} m \frac{d^2 x}{dt^2} &= F_{frX} + U_{ctrX} + F_{relX} + F_{rndX}, \\ m \frac{d^2 y}{dt^2} &= F_{frY} + U_{ctrY} + F_{relY} + F_{rndY}. \end{aligned} \quad (12)$$

Подводя итог, выделим некоторые существенные ограничения рассматриваемой нами модели.

- Мы рассматриваем дрон как материальную точку, т.е. отсутствует учет моментов и собственного вращения дрона.
- Мы предполагаем, что дрон может двигаться в произвольном направлении с одинаковыми динамическими характеристиками.
- Мы разделяем внешнее управляющее воздействие, необходимое для целеполагающих задач, стоящих перед дроном (роем) и внутреннего взаимодействия, обеспечивающего взаимную навигацию дронов. При этом все управляющие команды (всех видов) выражаются в виде некоторых виртуальных сил, действующих на дрон, а сам дрон движется под действием результирующей от всех воздействий.
- Воздействие сил взаимодействия определяется внутренней программой дрона и зависит от используемых датчиков, программно-аппаратных методов идентификации соседей и динамических характеристик дрона.
- Взаимодействующие дроны находятся на одной высоте.

Рассмотрим правила взаимодействия двух дронов более подробно. На рис. 1 выделены следующие области.

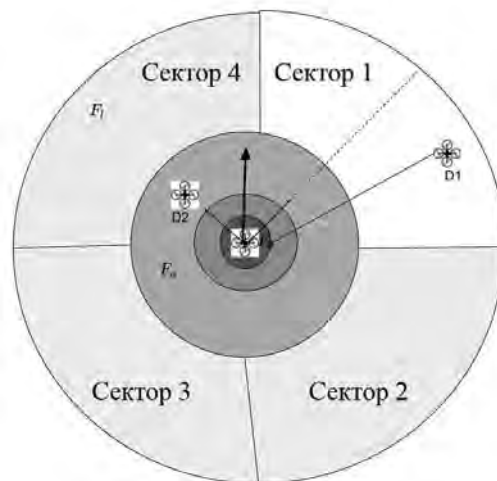


Рис. 1. Области видимости датчиков дрона

Светло-серым цветом обозначена область, в которой датчики могут определить наличие дрона с точностью до сектора в $1/4$ окружности. Разделение на 4 сектора — это минимально возможное количество секторов, которое дает возможность собрать рой. В этой области действуют силы дальнего притяжения в рамках сектора.

Белым цветом выделен один из секторов, в который попал дрон-сосед ($D1$). При его нахождении внутри сектора сила взаимодействия определяется аналогично силе дальнего притяжения, действие этой силы определяется особенностью приемных датчиков таким образом, что вектор этой силы направлен вдоль пунктирной линии, проходящей через центр окружности этого сектора. Все секторы равнозначны, т.е. у дрона нет выделенного направления, и если в нескольких секторах есть дроны-соседи, то на дрон действует равнодействующая, пропорциональная яркости излучения в данном секторе (так как используется простейший фотоэлемент, то мы будем считать, что излучение от нескольких дронов в рамках одного сектора будет складываться). Таким образом, один, но находящийся близко дрон будет более значим для системы взаимной навигации, чем несколько, но находящихся на удалении.

Средне-серым цветом обозначена область радиуса r_{a2} , в которой датчики точно определяют расположение и скорость дронов относительно друг друга. В этой области действуют датчики визуального контроля, позволяя определять относительные (относительно положения дрона) координаты соседа. В нашем примере в этой области находится дрон $D2$. На основании полученной информации мы имеем возможность моделировать силы ближнего притяжения, действующие по направлению к центру масс соседа, а также оценить относительную скорость дрона.

Темнее среднесерого обозначена область радиуса r_{a1} , в которой дроны могут находиться неограниченно долго и не взаимодействовать друг с другом.

Темно-серым цветом обозначена область радиуса r_d , характеристики датчиков в которой аналогичны описанным для зоны, обозначенной серым цветом. В эту область заходить соседним дронам не рекомендуется, так как это чревато столкновением (так называемая зона безопасности). Внутри этой зоны действуют силы отталкивания.

Стрелками обозначены виртуальные управляющие силы взаимодействия, *выделенная стрелка* — их результирующая.

Для определения значений и характера сил при взаимодействии дронов будем использовать аналог потенциала Леннарда–Джонса, используемого для описания межмолекулярного взаимодействия [6]. Так, согласно описанию потенциала Леннарда–Джонса, области притягивания и отталкивания определены через воздействия соответствующих сил:

$$\vec{F}_{rel} = \vec{F}_d + \vec{F}_a + \vec{F}_l.$$

Их области воздействия показаны на рис. 2.

Сила отталкивания (desire force) дронов друг от друга согласно аналогии будет задаваться следующим образом:

$$\vec{F}_d = k_d \frac{\vec{r}}{r^5} \mathbb{1}_d(r^2),$$

где k_d — постоянный коэффициент взаимодействия при отталкивании, $\vec{r}$ — вектор от текущего дрона как точки отсчета до соседа, взаимодействие с которым мы рассматриваем. Оба дрона в данной модели — материальные точки. Приведем индикаторную функцию расстояния $\mathbb{1}_d$, определяющую наличие силы отталкивания дронов друг от друга, если они сильно приблизились:

$$\mathbb{1}_d(r^2) = \begin{cases} 1, & r^2 < r_d^2, \\ 0, & r^2 \geq r_d^2. \end{cases} \quad (13)$$

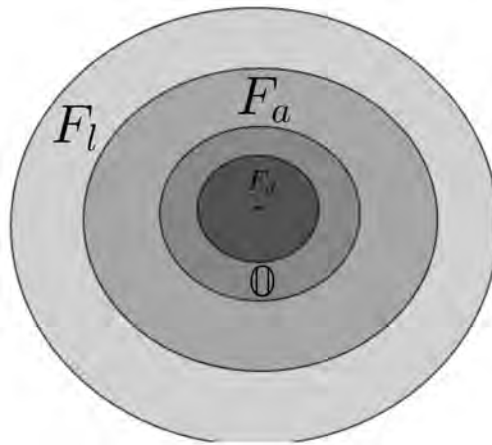


Рис. 2. Управляющие силы в зависимости от расстояния до соседа:

- F_l — сила, отвечающая за притяжение дрона на дальних расстояниях;
- F_a — сила, отвечающая за притяжение дрона на средних расстояниях;
- 0 — область отсутствия взаимодействия на ближне-средних расстояниях;
- F_d — сила отвечающая за отталкивание дронов друг от друга на ближних расстояниях

Другими словами, сработали датчики сближения или, в другом варианте конфигурации дрона, правила предупреждения столкновений.

Сила $\vec{F}_d$ виртуальна, т.е. она является результатом работы специального программно-аппаратного модуля, отвечающего за взаимодействие дронов. Она предназначена для двух целей:

- 1) избегание столкновений;
- 2) определение минимально допустимого сближения дронов внутри роя.

В общем случае, эти две цели могут быть реализованы разными правилами, но в данном случае действует одно правило.

Необходимо отметить, что на баланс притягивания и отталкивания в случае работы Леннарда–Джонса влияет разность степеней зависимости от расстояния, которая должна быть не менее 2, чтобы у дронов не было возможности с “разгона” при притягивании пролететь всю область безопасности и врезаться друг в друга. Сила притяжения дронов (attraction force) друг к другу на “среднем” расстоянии:

$$\vec{F}_a = k_a \frac{\vec{r}}{r^3} \mathbb{1}_a(r^2). \quad (14)$$

Здесь k_a — постоянный коэффициент взаимодействия при притягивании, $\vec{r}$ — вектор координат соседа, где текущий дрон является точкой отсчета, взаимодействие с которым мы рассматриваем,

$$\mathbb{1}_a(r^2) = \begin{cases} 1, & r_{a1}^2 < r^2 < r_{a2}^2, \\ 0, & r^2 \geq r_{a2}^2 \text{ или } r^2 \leq r_{a1}^2, \end{cases} \quad (15)$$

$\mathbb{1}_a$ — индикаторная функция, определяющая наличие силы взаимодействия при притяжении дронов друг к другу, если они находятся на определенном расстоянии, эта функция аналогична функции используемой при описании силы “отталкивания”, только она действует на средних расстояниях. Данная сила предназначена для обеспечения структурной сборки роя.

Сила притяжения дронов (long distance force) друг к другу на “большом расстоянии”:

$$\vec{F}_l = k_l \vec{r} \mathbb{1}_l(r^2),$$

где k_l — постоянный коэффициент взаимодействия при дальнем притягивании, $\vec{r}$ — единичный вектор определенного углового сектора, сектор определяется наличием фотоэлемента, направленного в определенную сторону, чувствительный элемент которого может “поймать” сигнал в угловом секторе размером $\pi/2$ (четверть круга) и определить “светимость”. В силу физических особенностей данного датчика сигнал от него пропорционален сумме “светимостей” от нескольких дронов. Необходимо отметить также, что размер сектора выбран таким образом, чтобы обеспечивать приемлемую взаимную навигацию при минимальной стоимости. В процессе работы были также опробованы варианты с размером сектора $\pi/4$ и $\pi/6$, но существенного выигрыша при стабилизации роя не было.

Приведем индикаторную функцию, определяющую наличие силы взаимодействия при притяжении дронов на дальних расстояниях в рамках сектора,

$$\mathbb{1}_l(r^2) = \begin{cases} 1, & r_{a2} < r^2 < r_l^2, \\ 0, & r^2 \geq r_l^2 \text{ или } r^2 \leq r_{a2}^2. \end{cases} \quad (16)$$

Если дроны совсем далеко, вне радиуса r_l , то они друг друга не “видят”, а если они оказались внутри радиуса, то максимум, что можно определить — это то, что в данном секторе наблюдается некая “светимость”, никакой информации ни о расстоянии до соседей, ни об их относительных координатах, дрон определить не может.

Таким образом, при взаимодействии двух дронов в каждый момент времени на дрон может действовать только одна внутренняя сила в зависимости от расстояния между дронами. Если дронов несколько, то силы их воздействия складываются (а часто и взаимно компенсируются).

В то же время данная модель позволяет снять некоторые ограничения, обычно налагаемые на подобные системы.

1. В нашей модели мы исходим из предположения, что нам не доступна координатная навигация, а дрон с разной степенью точности “видит” только своих соседей. В соответствии с ними работает программа внутреннего управления, подающая команды на контроллер соответствующих двигателей, воздействие которых можно свести к воздействию управляющих сил.
2. Данная модель роя не ограничена законами сохранения (энергии и импульса), однако симметрия пространства и симметрия законов управления дает нам возможность говорить о сохранении положения центра масс роя в отсутствии U_{ctr} .

5. Описание состояние роя на основе динамического имитационного моделирования. На основании данной модели самого дрона, модели датчиков и модели управляющих сил (правил) мы построим многоагентную систему роя, где каждый агент взаимодействует с другими через модель окружающей среды.

Запишем состояние роя с использованием матричного подхода. Пусть матрицы

$$X(t) = \begin{bmatrix} x_1(t) \\ \vdots \\ x_n(t) \end{bmatrix}, Y(t) = \begin{bmatrix} y_1(t) \\ \vdots \\ y_n(t) \end{bmatrix} \quad (17)$$

задают соответствующие координаты каждого дрона в рое в статической системе координат в каждый момент времени. Тогда $\vec{R}(t) = [X(t); Y(t)]$ будем называть гипервектором координат роя. Таким образом, попарные относительные координаты дронов будут представлять собой квадратную матрицу:

$$\begin{aligned} \tilde{X} &= E \circ X - X^T \circ E, \\ \tilde{Y} &= E \circ Y - Y^T \circ E, \end{aligned} \quad (18)$$

где $E = (1)_{n \times n}$ — единичная матрица в смысле произведения Адамара. Здесь столбцы $\tilde{X}_i, \tilde{Y}_i$ — координаты группы относительно i -го дрона. Приведем пример матриц отклонения $\tilde{X}, \tilde{Y}$ для

случая трех дронов:

$$\begin{aligned} \begin{bmatrix} 0 & x_1 - x_2 & x_1 - x_3 \\ x_2 - x_1 & 0 & x_2 - x_3 \\ x_3 - x_1 & x_3 - x_2 & 0 \end{bmatrix} &= \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \circ \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} - \begin{bmatrix} x_1 & x_2 & x_3 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}, \\ \begin{bmatrix} 0 & y_1 - y_2 & y_1 - y_3 \\ y_2 - y_1 & 0 & y_2 - y_3 \\ y_3 - y_1 & y_3 - y_2 & 0 \end{bmatrix} &= \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \circ \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} - \begin{bmatrix} y_1 & y_2 & y_3 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}. \end{aligned} \quad (19)$$

Тогда попарные расстояния между дронами будут описываться как симметричная матрица вида $R^2 = \tilde{X}^2 + \tilde{Y}^2$. Для случая трех дронов

$$\begin{bmatrix} 0 & r_{12}^2 & r_{13}^2 \\ r_{21}^2 & 0 & r_{23}^2 \\ r_{31}^2 & r_{32}^2 & 0 \end{bmatrix} = \begin{bmatrix} 0 & (x_1 - x_2)^2 & (x_1 - x_3)^2 \\ (x_2 - x_1)^2 & 0 & (x_2 - x_3)^2 \\ (x_3 - x_1)^2 & (x_3 - x_2)^2 & 0 \end{bmatrix} + \begin{bmatrix} 0 & (y_1 - y_2)^2 & (y_1 - y_3)^2 \\ (y_2 - y_1)^2 & 0 & (y_2 - y_3)^2 \\ (y_3 - y_1)^2 & (y_3 - y_2)^2 & 0 \end{bmatrix}. \quad (20)$$

Введя булеву функцию по аналогии с индикаторной функцией, описанной выше, получим

$$\mathbb{1}_{r_1, r_2}(R^2) = \begin{cases} 1, & r_1^2 < r_{ij}^2 < r_2^2, \\ 0, & r_{ij}^2 \geq r_2^2 \text{ или } r_{ij}^2 \leq r_1^2. \end{cases} \quad (21)$$

Здесь r_1, r_2 — граничные радиусы, внутри которых действует одна из сил взаимодействия $r_1 < r_2$, r_{ij} — это расстояние от дрона $\mathbf{i}$ до дрона $\mathbf{j}$. Таким образом состояние роя в абсолютных координатах будет описываться уравнениями

$$\begin{aligned} m \frac{d^2 X}{dt^2} &= F_{frX} + F_{ctrX} + F_{relX} + F_{rndX}, \\ m \frac{d^2 Y}{dt^2} &= F_{frY} + F_{ctrY} + F_{relY} + F_{rndY}. \end{aligned} \quad (22)$$

В силу линейности изменения матрицы абсолютных и относительных координат в каждый момент времени получаем

$$\begin{aligned} \frac{d^2 X}{dt^2} &= \frac{d^2 \tilde{X}}{dt^2}, \\ \frac{d^2 Y}{dt^2} &= \frac{d^2 \tilde{Y}}{dt^2}. \end{aligned} \quad (23)$$

Сила взаимодействия F_{rel} такая же, как для одного дрона, но состоит из гипервекторов. По аналогии с заданным выше гипервектором координат роя мы будем говорить о гипервекторе сил, действующих на рой:

$$F_{rel} = F_d + F_a + F_l. \quad (24)$$

В проекции на оси координат:

$$\begin{cases} F_{relX} = F_{dX} + F_{aX} + F_{lX}, \\ F_{relY} = F_{dY} + F_{aY} + F_{lY}, \end{cases} \quad (25)$$

$$\vec{F}_d = k_d \frac{\vec{r}}{r^5} \mathbb{1}_d(r^2) \Rightarrow \begin{cases} \tilde{F}_{d,X} = k_d \tilde{X} \circ R^{-5} \circ \mathbb{1}_d(R^2), \\ \tilde{F}_{d,Y} = k_d \tilde{Y} \circ R^{-5} \circ \mathbb{1}_d(R^2), \end{cases} \quad (26)$$

где R^{-5} — псевдообратная матрица R^5 . При этом квадратные матрицы взаимного влияния дронов $\tilde{F}_{dX}, \tilde{F}_{dY}$ при отталкивании преобразуются в столбцы

$$\begin{aligned} F_{d,X} &= \langle \tilde{F}_{d,X} \rangle_1, \\ F_{d,Y} &= \langle \tilde{F}_{d,Y} \rangle_1. \end{aligned} \quad (27)$$

Аналогично описывается сила притяжения дронов на близком и на дальнем расстоянии. Отметим, что в данном случае все матрицы симметричны и соответственно симметричны их адамаровы произведения, а свертка по столбцу будет эквивалентна свертке по строке:

$$\vec{F}_a = k_a \frac{\vec{r}}{r^3} \mathbb{1}_a(r^2) \Rightarrow \begin{cases} \tilde{F}_{a,X} = k_a \tilde{X} \circ R^{-3} \circ \mathbb{1}_a(R^2), \\ \tilde{F}_{a,Y} = k_a \tilde{Y} \circ R^{-3} \circ \mathbb{1}_a(R^2), \end{cases} \quad (28)$$

где R^{-3} — псевдообратная матрица R^3 . Аналогично, $\tilde{F}_{d,X}$, $\tilde{F}_{d,Y}$ — квадратные матрицы взаимного влияния, которые при отталкивании дронов преобразуются к виду

$$\begin{aligned} F_{a,X} &= \langle \tilde{F}_{a,X} \rangle_{\downarrow}, \\ F_{a,Y} &= \langle \tilde{F}_{a,Y} \rangle_{\downarrow}. \end{aligned} \quad (29)$$

6. Выводы. Сравнение двух подходов. Как видно, описание движения единичного дрона при использовании произведения Адамара практически эквивалентно описанию движения всего роя, что очень упрощает как реализацию модели, так и эксперименты с воздействием дронов друг на друга.

С другой стороны, использование такого подхода может накладывать, если не ограничения, то, по крайней мере, может усложнить моделирование в следующих случаях:

- если описание законов управления невозможно свести к воздействию разнонаправленных сил на дрон в каждый момент времени;
- если воздействие дронов друг на друга не является одинаковым, т.е. в рое существуют выделенные (главные) дроны. Данный аспект не исследовался и находится вне рамок данной работы, но наш подход все же позволяет моделировать иерархические модели дронов, если заменить индикаторную функцию или единичную матрицу в формулах относительных координат на весовые матрицы;
- однако самым важным ограничением будет введение таких законов взаимодействия между дронами, при которых воздействие нескольких дронов на один, не аддитивно, однако и в этом случае изменение введенной ранее операции свертки на другую операцию, действующую в рамках описанной алгебры даст возможность обойти это ограничение.

При соблюдении указанных нами ограничений мы можем наблюдать полную идентичность результатов, полученных в расчетах с помощью многоагентной и системно-динамической моделей (см. рис. 3, 4).

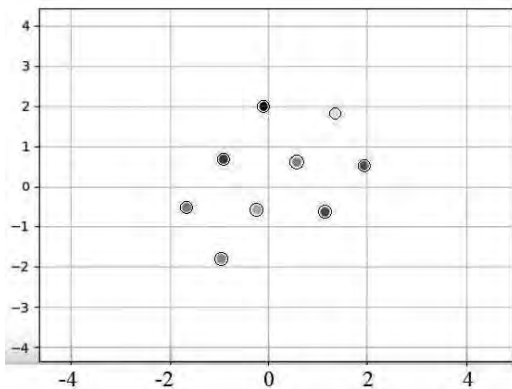


Рис. 3. Динамическая модель

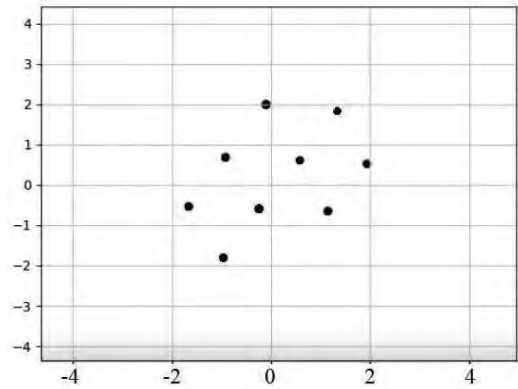


Рис. 4. Многоагентная модель

На рис. 3,4 показано состояние роя через 1 мин 10 с после начала моделирования. Шаг по времени составляет 0.05 с. Начальные состояния совпадают.

Таким образом поставленная нами цель, а именно, получение описания движения роя как единого объекта была достигнута. В процессе разработки модели мы получили методику преобразования частной модели дрона в динамическую модель роя, что было продемонстрировано на приведенном примере перехода от модели одиночного дрона к модели на базе алгебры Адамара.

Одним из важных технических достоинств данного подхода является возможность использовать готовые реализации указанных выше операций на базе произведения Адамара и введенных нами расширений, так как они являются стандартом операций машинного обучения в большинстве современных математических пакетов.

Для сравнения производительности при применении данного подхода нами была реализована описанная выше модель роя на базе многоагентного взаимодействия, а также в виде динамической (матричной) имитационной модели, причем последняя была протестирована в однопоточном, многопоточном вариантах, а также варианте с использованием GPU.

Характеристики решателя для всех реализаций были следующие: состояние роя рассчитывалось в течение внутренних 200 с с шагом 0.05 с, т.е. было сделано 4000 итераций. Опишем подробнее используемые при вычислениях ресурсы.

MultiAgent — мультиагентное моделирование проводилось на шестиядерном процессоре в режиме использования 6 активных ядер для вычислений. Обмен сообщениями производился в режиме синхронного распределения памяти.

Single Thread CPU — динамическое имитационное моделирование на базе алгебры Адамара. Проводилось на том же компьютере с 6 ядрами, но при использовании одного ядра и стандартных библиотек матричных вычислений.

Multi Thread CPU 6 cores — та же самая динамическая модель, но вычисления производились на базе многопоточного вычислителя TensorflowCPU. Использовались все ядра.

Multi Thread CPU 26 cores — модель аналогична предыдущей, но с использованием более мощного процессора. Данная проверка была сделана для подтверждения предположения, что производительность возрастает линейно от количества ядер процессора.

GPU Nvidia 1070 — динамическая модель, аналогичная предыдущим вариантам, но при вычислениях использовалась реализация вычислителя на базе TensorflowGPU.

Моделирование роя как многоагентной системы показало результат в 6 раз медленнее, чем динамическое моделирование в однопоточном режиме, что может быть критически важно для работы в режиме реального времени.

Сравнительные затраты по времени для модели роя, описанного в качестве примера применения алгебры Адамара (с одинаковыми характеристиками решателя)

Количество дронов	Multy Agent, с	Single Thread CPU, с	Multi Thread CPU 6 cores, с	Multi Thread CPU 26 cores, с	GPU Nvidia 1070, с
100	2110	313	40.2	20	38
500	9641 (3 ч)	1489	196	48	217
1000	20543 (6 ч)	3230 (1 ч)	423	107	413

Как видно из таблицы, рост производительности при задействовании разного количества ядер почти линейный, что говорит о корректности реализованных алгоритмов с точки зрения обработки матриц. Другими словами, предложенный алгоритм описания модели позволил избавиться

от поэлементного вычисления состояния каждого дрона и перейти к высокоэффективным матричным операциям.

Также мы провели эксперименты по расчету поведения роя при использовании GPU в тех же условиях. К сожалению, производительность относительно старых веерных процессоров Pascal based GPU игрового исполнения, сравнима, в данном случае, с использованием 6-ядерного процессора. При этом использование данного подхода осложняется проблемами с отладкой программного обеспечения под GPU. Однако при использовании более мощных современных GPU, ориентированных на высокопроизводительные вычисления (Titanium edition или Tesla) мы получим выигрыш по сравнению с использованием процессоров. Также были произведены тесты масштабирования, которые показали линейный рост времени моделирования относительно роста количества объектов моделирования (дронов).

К существенным недостаткам данного подхода мы бы отнесли следующую особенность: разработка программы для агента фактически означает разработку программы для устройства, при использовании предложенного подхода программу для дрона необходимо будет написать отдельно, но это, как сказано выше, компенсируется возрастанием производительности при моделировании роя почти в 50 раз.

СПИСОК ЛИТЕРАТУРЫ

1. Magnus J.R., Neudecke H. Matrix differential calculus with applications to simple, Hadamard and Kronecker product // J. of Math. Psychol. 1975. P. 3–228.
2. Maria A. Introduction to modeling and simulation // Proceedings of the 29th Conference on Winter Simulation. USA, 1997. P. 7–14.
3. Davis C. The norm of the Schur product operation // Numerische Mathematik. 1962. P. 343–344.
4. Фаддеев Д.К., Фаддеева В.Н. Вычислительные методы линейной алгебры. Л.: Наука, 1975.
5. Girgidov R.A., Khakimov T. Self-organization of a group of drones in the conditions of absence of data on the exact location // Algorithms and Solutions Based on Computer Technology. Lecture Notes in Networks and Systems. Vol. 387. Springer, 2022. P. 223–233.
6. Jones J.E. On the determination of molecular fields. I. From variation of the viscosity of a gas with temperature // Proceedings of The Royal Society A: Mathematical, Physical and Engineering Sciences. Vol. 106. 1924. P. 441–462.

Поступила в редакцию 19.01.23

Одобрена после рецензирования 15.03.23

Принята к публикации 15.03.23