УДК 629.5.061; 629.5.072
DOI: 10.37102/1992-4429_2024_49_03_05
EDN: OHDAIT

АВТОНОМНОЕ СЛЕДОВАНИЕ БЕЗЭКИПАЖНОГО СУДНА ПО ТРАЕКТОРИИ И ПРЕДОТВРАЩЕНИЕ СТОЛКНОВЕНИЙ

М.П. Фархадов, А.Н. Сорокин, И.Ф. Жидомиров, С.С. Керимов

В данной статье предлагается метод решения проблемы навигации безэкипажного судна в условиях высокой неопределенности. Основная цель – заставить агента обучения с подкреплением выучить алгоритм, позволяющий безэкипажному судну следовать по определенной траектории, избегая столкновений с другими объектами. Маневрирование безэкипажного судна в предложенных условиях является основной темой данной работы. В ходе исследования было изучено несколько сценариев со статическими и динамическими объектами. Обучение агента проводилось с помощью алгоритмов, не требующих моделирования и не связанных с алгоритмом управления напрямую. Процесс обучения был разделен на несколько частей, в которых мы экспериментировали с элементами подхода метаобучения для достижения устойчивости поведения агента.

Ключевые слова: следование по траектории, маневрирование, машинное обучение, обучение с подкреплением, свободная модель, избегание препятствий

Введение

При одинаковых условиях и обстоятельствах плавание судов в открытых водах не является столь же опасной и сложной задачей, как плавание в зонах с высокой плотностью движения или в районах с большим количеством статических препятствий. При возникновении риска столкновения в открытых водах существует несколько вариантов действий в данной ситуации. В зонах с ограниченным движением иногда существует единственный допустимый маневр, который необходимо предпринять в конкретный момент. Здесь «допустимый» означает не единственно возможный, а тот, который является приемлемым с учетом текущих обстоятельств. Более того, в данном исследовании мы не ограничиваем поведение агента какими-либо установленными IMO (Международной морской организацией) навигационными правилами, такими как ColReg (Правила предупреждения столкновений судов в море). Мы анализируем способность агента реагировать на условия высокой неопределенности при наличии лишь частичной информации об окружающей среде. Желаемая адаптивная методика должна придерживаться строгого следования по траектории в ситуации нулевого риска столкновения и отклоняться от первоначального поведения, чтобы смягчить или даже устранить риск столкновения, когда он возникает. Методика воплощается глубокой нейронной сетью, параметры которой оптимизируются на основе функции потерь, полученной алгоритмом обучения с подкреплением (ОП). Методика без привязки к конкретной модели позволяет не обращать внимания на модель окружающей среды и оценивать текущую ситуацию исключительно по сигналу вознаграждения.

Контекст

Агент представляет собой модель судна с тремя степенями свободы (3-DOF), существующего в среде, описываемой плоским миром. Путь — это либо прямая, либо кривая линия, соединяющая две последовательные точки. Основная мотивация агента – следовать по пути между точкой отправления (ТО) и точкой прибытия (ТП). При отклонении от этого пути агент подвергается незначительному наказанию. Еще одно ограничение – максимальная скорость, когда она превышает определенный порог, агент штрафуется. Наконец, самое важное наказание налагается на агента, если он сталкивается с каким-либо препятствием. Единственная положительная награда начисляется каждый раз, когда агент приближает-

ся к цели. Предусмотрено два типа препятствий. Статические – все виды целей, которые сохраняют свое положение в окружающей среде в течение одного эпизода. Динамические – цели, которые могут развивать определенную скорость и самостоятельно перемещаться в соответствии с собственной логикой. Агент может одновременно наблюдать за ограниченным количеством статических и динамических препятствий. Видимость ограничена полем зрения, все цели за пределами поля зрения считаются ненаблюдаемыми и не включаются в вектор состояния агента.

Похожие работы

Автономная навигация любого типа — это область, которая постоянно развивается и в которой в последнее время было проведено множество исследований [1]. Движение беспилотных летательных аппаратов на открытом и в закрытом пространстве, визуальная, подводная и все другие виды навигации активно развиваются в текущий момент и невозможны без автономной навигации. Безусловно, наибольший вклад в эту область вносит автономное вождение, так как основные принципы алгоритмизации остаются прежними независимо от среды.

Последние работы по автономной навигации судов в различных условиях вносят важный вклад в развитие всей технологии автономных судов. Мы рассмотрим различные подходы, применяемые для автономной навигации в морской области. Авторы в [2, 3] предложили алгоритм Deep Deterministic Policy Gradient (DDPG) – алгоритм, основанный на глубоком ОП, для навигации судна по узкому каналу. Агент должен изучить законы управления для безопасной и эффективной навигации в канале без препятствий. Интерес алгоритма DDPG для данной работы заключается в том, что он может быть использован для построения алгоритма непрерывного решения, который может быть использован для построения закона управления движителями и рулями.

В [2] Zhao Luman и др. описали глубокий ОП-агент для следования по траектории и предотвращения столкновений на основе алгоритма Proximal Policy Optimization (PPO). Согласно статье, агент должен научиться наиболее безопасному и экономичному поведению по обходу препятствий при взаимодействии с движущимися препятствиями, и это поведение должно соответствовать ColReg.

В работах [5 ,6, 7, 8] исследовали проблему предотвращения столкновения нескольких кораблей в условиях неизвестной среды. Они предложили комбинированный метод обучения на основе асинхронного алгоритма «актор-критик» (АК), нейронной сети с длинной кратковременной памятью (LSTM) и Q-обучения для решения проблемы низкой эффективности безмодельного ОП. Их метод использует Q-learning для адаптивного принятия решений между инверсным контроллером на основе модели LSTM и безмодельного алгоритма АК.

Еще один интересный подход был продемонстрирован в [9, 10]. Был предложен иерархический мультиагентный алгоритм ОП для уменьшения заторов, повышения безопасности навигации и предотвращения столкновений в оживленных и географически тесных районах судоходства. Они используют природу коллективных взаимодействий между агентами для разработки градиентного подхода, который может масштабироваться на более крупные проблемы. Сотни агентов действуют в децентрализованной среде, где агенты наблюдают за движением только в своем районе. Многоагентный метод распределения кредитов точно определяет вклад каждого мета-действия в общую цель управления движением для уменьшения высокой дисперсии оценок градиента [11, 12].

Описание модели

Окружающая среда

В нашем исследовании мы использовали среду, основанную на проекте gncgym https://github.com/haakonrob/gncgym. Она представляет собой плоский водный мир, в котором действует агент, взаимодействующий с другими объектами.

Для данной работы нам необходимо описать среду в терминах ОП. Основной набор параметров состоит из агента, пространства состояний, пространства действий, функции вознаграждения и функции перехода.

- Пространство состояний: мы предполагаем, что пространство состояний частично наблюдается агентом. Как было описано ранее, мы считаем видимым для агента только постоянное количество объектов. Если препятствие находится вне зоны видимости агента, оно не будет включено в вектор пространства состояний. В нашем случае состояние состоит из собственного положения судна { , }, скорости судна над поверхностью, курса, ошибки пересечения траектории, ошибки ограничения скорости, положения цели. Кроме того, в пространство состояний включены 4 вектора статических и динамических препятствий. Вектор статического препятствия описывается его положением {X, Y}. Вектор динами-

ческого препятствия состоит из его текущего положения, текущей скорости V и курса C.

- Пространство действий: пространство действий описывается двумя параметрами: тягой и углом поворота руля. Мы ограничили допустимые значения этих параметров некоторыми разумными пределами {0, 1} для тяги и {-1, 1} для угла поворота руля. На каждом шаге в среде агент совершает действие в заданных пределах.
- Функция вознаграждения: функция вознаграждения строится из нескольких параметров. Главным компонентом вознаграждения является расстояние до целевой точки, оно заставляет агента двигаться к интересующей его точке. Другими компонентами являются: ошибка отклонения от направления, которая мотивирует агента держаться как можно ближе к маршруту; ошибка ограничения скорости, которая заставляет агента не превышать физические ограничения.
- Функция перехода: вероятность того, что агент перейдет из одного состояния в другое, называется вероятностью перехода. Вероятность перехода из одного состояния в другое может быть выражена следующим образом: для марковского состояния из S[t] в S[t+1], т.е. любого другого состояния-преемника. Вероятность перехода состояния задается: вероятность перехода заложена в среде и обусловлена моделью сосуда и законами движения твердого тела в жидкости.

Модель судна

Существует абстрактная упрощенная модель судна с тремя степенями свободы, которая выступает в качестве агента в нашем окружении, поскольку любое судно можно аппроксимировать как модель с 3-DOF. Мы можем описать векторы состояния как $\nu = [u, v, r]^T$ и $\eta = [x, y, \psi]^T$ для линейной скорости в рамках тела [10]. Для надводного судна мы имеем общую кинематику жесткого тела. Кинематика жесткого тела может быть описана уравнением:

$$\eta = R(\psi)\upsilon,$$

где R – матрица поворота, преобразующая положения из неподвижной системы координат Земли в неподвижную систему координат судна, а υ – скорость судна. Динамическое уравнение для судна может быть выражено в следующей форме:

$$M(\dot{\upsilon}) + C(\upsilon)\upsilon + D(\upsilon)\upsilon = \tau,$$

где $M = M_{RB} + M_A$ – матрица масс, состоящая из массы жесткого тела и гидродинамической добавочной массы; $C(\upsilon) = C_{RB} + C_A(\upsilon)$ – матрица, содержащая гидродинамическую и кориолисову центростремительные матрицы жесткого тела; $D(\upsilon) = D_L + D_{NL}(\upsilon)$ – матрица нелинейного демпфирования, представляющая собой комбинацию линейного и нелинейного демпфирования; υ – относительная скорость судна по отношению к океанскому течению. Здесь $\nu = R(\psi)$.

Это можно упростить в выражение:

$$M\dot{\upsilon}_r = -C(\upsilon)\upsilon - (C(\upsilon)\upsilon + D(\upsilon)\upsilon) + \tau,$$

которое может быть перестроено для использования контролируемого изменения скорости судна как:

$$\dot{\upsilon} = M(-C(\upsilon)\upsilon - (C(\upsilon)\upsilon + D(\upsilon)\upsilon) + \tau).$$

Наша модель имеет только два управляющих воздействия: угол поворота руля и мощность акселератора. Управляющая сила τ может быть выражена как $\tau = [X_\delta\delta, Y_\delta\delta, N_\delta\delta]^T$, где Xδ, Yδ, Nδ – коэффициенты руля.

Предложенный метод

Мотивация

В основу предлагаемой работы легла первоначальная идея создания иерархической составной модели, которая может управляться агентом высокого уровня. Каждый элемент иерархии является задачей для отдельного агента нижнего уровня. Планирование прохода и предотвращение столкновений является важной частью рутинной работы морского транспорта и представляет собой отдельный уровень в данной системе.

Другой мотив – сравнить производительность нашего агента с известными алгоритмами динамического программирования, такими как A-star, при решении одной и той же задачи. Даже если высокая производительность детерминированных алгоритмов хорошо известна в задачах поиска кратчайшего расстояния для заданной среды. Наша ситуация осложняется динамически меняющейся средой высокого порядка, которая может быть сложной с вычислительной точки зрения.

Цель

Основной целью в данной работе было обучить ОП-агента следовать заданному маршруту между двумя точками. Кроме того, мы предполагаем наличие движущихся и статичных объектов, которые мы

рассматриваем как препятствия. Эта задача может быть разбита на поиск кратчайшего расстояния между двумя последовательными точками, учитывая, что мы не можем приблизиться к препятствиям ближе определенного порога.

Комбинацию препятствий в виде статичных тел можно рассматривать как лабиринт, из которого нужно выбраться, чтобы выполнить поставленную задачу. С другой стороны, динамические объекты следуют своей собственной логике и могут рассматриваться как препятствия или нерелевантные объекты. Постепенно изучая законы движения динамических объектов, агент оценивает динамически меняющуюся среду как статичный лабиринт на каждом конкретном шаге. Наконец, агент должен найти способ безопасно следовать по предложенному маршруту до целевой точки.

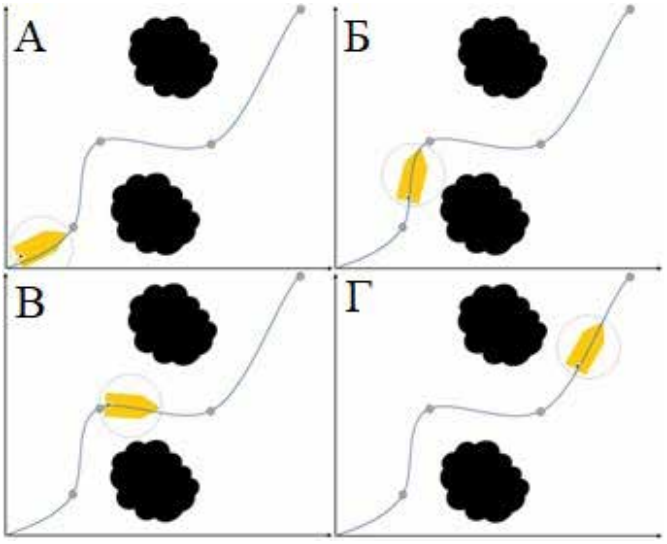

Рис. 1. Избежание столкновения со статическим препятствием

Метод

В отличие от процесса навигации реального судна, в нашем исследовании агенту уже предложен первоначальный план движения. Он должен следовать по этому маршруту до тех пор, пока не прибудет в конечную точку. На рис. 1 показано поведение судна, которое движется от начальной точки маршрута к конечной. Если оно идет прямо между этими двумя точками, то сталкивается с препятствием. Для безопасного прохождения этого маршрута ему необходимо маневрировать, тем самым изменяя свой первоначальный план движения.

Здесь мы не требуем от агента явного построения маршрута путем установки новых путевых точек. Вместо этого агент, наблюдающий за ситуацией, должен постоянно оценивать риск столкновения и действовать соответствующим образом. Можно представить, что он планирует новый маршрут, модифицируя его дополнительными путевыми точками, учитывая, что ситуация меняется, но, скорее всего, это не так. В этом и заключается вся идея ОП: с помощью функции вознаграждения мы объясняем агенту правила игры, по крайней мере то, как мы их понимаем. В этих терминах отклонение от маршрута для избегания столкновения и при этом приближение к точке назначения – главный принцип, который должен усвоить агент. Внешняя мотивация, выраженная в функции вознаграждения, заставляет его постоянно приближаться к точке назначения. Небольшой штраф мотивирует его держаться как можно ближе к маршруту, в то время как в случае столкновения он получит огромный штраф.

Ситуация усложняется, когда агент замечает динамическое препятствие или просто другое судно. На рис. 2 изображен один из возможных сценариев. Фрагмент «А» показывает первоначальное намерение агента избежать столкновения со статичным препятствием. В то время как он наблюдает присутствие другого судна поблизости, он начинает получать информацию об этом объекте (фрагмент «Б»), одновременно оценивая возможный риск столкновения, если они оба сохранят свой курс и скорость. С течением времени агент собирает все больше информации, и оценка риска столкновения становится все более точной. Фрагмент «В» изображает изменение намерений: в текущей ситуации агент оценил риск как неприемлемый и изменил маршрут.

Мы ни в коем случае не объясняем агенту, как именно действовать в каждой ситуации, не суще-

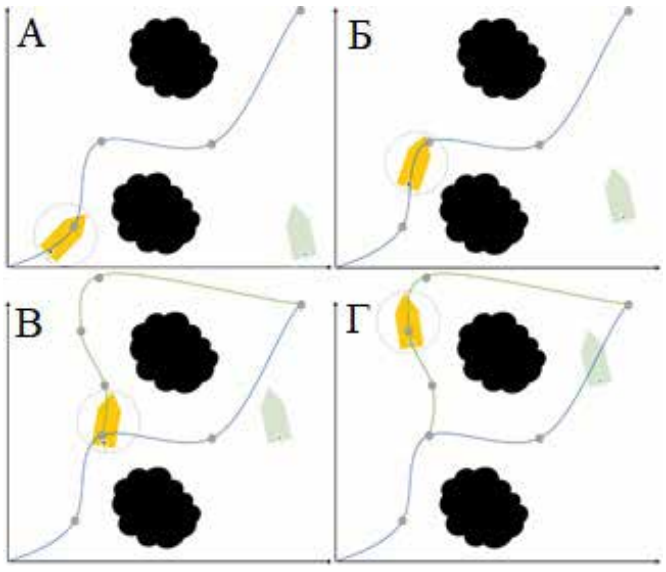

Рис. 2. Избежание столкновения с динамическим препятствием

ствует какого-либо алгоритма, охватывающего все возможные состояния, включая угловые случаи. Мы лишь приблизительно описали наше предложение в виде сигнала, который мы назвали функцией вознаграждения.

Алгоритм

Для нашего исследования мы решили выбрать алгоритм управления, сочетающий в себе преимущества обоих типов обучения. Агент представлен алгоритмом управления из семейства глубоких детерминированных градиентных алгоритмов DDPG. Кратко напомним некоторые основные понятия ОП.

Методы управления, основанные на ценностях, обучают детерминированному алгоритму, в котором агент при принятии решения выбирает действие по принципу максимизации награды за него. Основным преимуществом обучения вне алгоритма является возможность обучения агента на основе данных, полученных в результате применения различных алгоритмов управления. Эти алгоритмы очень эффективны при работе с данными.

В градиентных алгоритмах [13] методика, выученная агентом, является стохастической. Присваивание вероятности каждому дискретному действию или требование выборки из некоторого распределения для непрерывных действий обеспечивает исследование пространства состояний и действий. Хотя несколько действий имеют некоторую вероятность быть выбранными, агент не пропустит потенциально высокое вознаграждение, которое может быть проигнорировано в случае детерминированного алгоритма, выученного алгоритмами, основанными на ценности.

Недостатком такого поведения является завязка этих методик на алгоритмы управления. Они требуют свежих данных для каждого обновления и не могут повторно использоваться в дальнейшем, потому что значение оцениваемого действия должно быть произведено агентом недавно. В противном случае смещение будет расти и приведет к тому, что алгоритм никогда не сойдется. С другой стороны, несколько эпизодов, сгенерированных одним и тем же оптимальным алгоритмом, могут дать существенно различающиеся результаты. Чтобы смягчить несмещенную дисперсию, вызванную такой особенностью, необходимо большее количество выборок. Это объясняет, почему градиентные методы алгоритмов управления [11, 12] имеют худшую сложность выборки, чем методы, основанные на значениях.

Детерминированный градиент алгоритма (DPG) вместо этого моделирует алгоритм управления как детерминированное решение: a= μ(S). Мы можем рассматривать детерминированный алгоритм как частный случай стохастической методики, когда распределение вероятностей содержит только одно экстремальное ненулевое значение для одного действия. Было показано [1], что стохастическая методика $\pi\mu\theta,\sigma$ перепараметризуется детерминированным алгоритмом μθ и вариационной переменной σ, при этом стохастическая методика в конечном итоге эквивалентна детерминированной при $\sigma = 0$.

Теорема о градиенте детерминированного алгоритма управления может быть применена с обычным фреймом градиента алгоритма управления. Интегрируя его в актор-критик, мы получаем:

$$\nabla_\theta J(\theta) = E_{s\sim p\mu}[\nabla_{\theta\mu\theta}(s) * \nabla_\alpha Q(s,a) \mid \alpha = \mu_\theta(s)].$$

Это обозначение означает, что градиент Q-значения берется при $a = \mu_\theta(s)$. Мы пытаемся найти влияние изменения параметров актора θ на максимизацию Q-значения. $\nabla_{\theta\mu\theta}(s)$ зависит только от параметризованного актора. Термин $\nabla_\alpha Q(s,a)$ является своего рода критиком, показывающим актору, в каком направлении следует изменить алгоритм управления: в сторону действий, связанных с большим вознаграждением. Это основная идея DPG, используемая в семействе алгоритмов, которые мы будем применять. Что касается критика, то он обучается с помощью Q-обучения и целевых сетей.

$$J_\theta =$$

$$= E_{s\sim p\mu}\left[\left(r(s,\ a,\ s`) + \gamma * Q_{\varphi`}(s`, \mu_{\theta`}(s`)) - Q_\varphi(s,a)\right)^2\right]$$

Целевые сети медленно отслеживают обученные сети, обновляя свои параметры после каждого обновления обученной сети с помощью скользящего среднего для актора и критика.

$$\theta` = \tau * \theta + (1 - \tau) * \theta`$$

Пока τ намного меньше 1, веса целевых сетей обновляются медленно, обеспечивая определенную стабильность обучения Q-значениям.

Когда методика детерминирована, в процессе обучения, скорее всего, она находит «лучшее» действие, по крайней мере для некоторых состояний, потенциально пропуская «неинтересные» действия, которые могут привести к более высокой награде в долгосрочной перспективе. Чтобы обеспечить исследование среды, мы добавим к детерминированным действиям некоторое стохастическое количество шума:

$$a_t = \mu_\theta(s_t) + \xi$$

Архитектура алгоритма DDPG представлена на схеме (рис. 3).

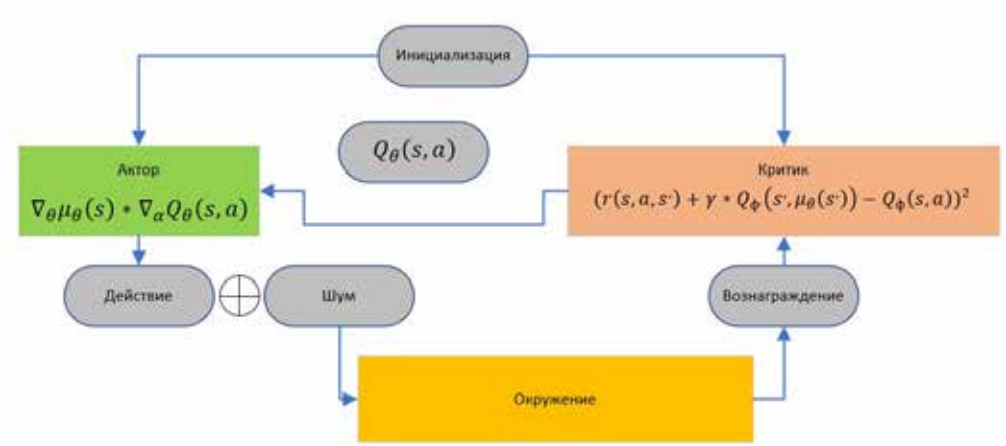

Рис. 3. DDPG в акторно-критическом фрейме

Однако прямая максимизация объективной функции, вероятно, будет нестабильной и будет страдать от переоценки функции Q-значения. Эта переоценка может распространиться через итерации обучения и негативно повлиять на алгоритм управления. Twin Delayed Deep Deterministic (TD3) [1] применил пару трюков к DDPG, чтобы предотвратить переоценку функции ценности.

Обрезанное двойное Q-обучение: две независимые сети производят оценку выбора действия и Q-значения. В DDPG даны два детерминированных актора с двумя соответствующими критиками. Поскольку методика меняется медленно, эти две сети могут быть слишком похожи, чтобы принимать независимые решения. Обрезанное двойное Q-обучение сохраняет только минимальную оценку из двух, чтобы не допустить ошибки недооценки, которую трудно распространить в процессе обучения.

$$y = r + \gamma * \min_{i=1,2} Q_{wi}\left(s`, \mu_{\theta 1}\left(s`\right)\right),$$

$$y = r + \gamma * \min_{i=1,2} Q_{wi}\left(s`, \mu_{\theta 2}\left(s`\right)\right).$$

Задержка обновления сетей целей и алгоритма: обновления алгоритма управления и ценностей фактически взаимозависимы в обобщенных итерационных системах алгоритмов управления, таких как актор-критик: когда методика сильно неоптимальна, оценки ценностей расходятся из-за переоценки, и методика будет расходиться, если сама оценка ценностей неточна. Чтобы обеспечить стабильное обучение и уменьшить дисперсию, нам нужно обновлять алгоритм управления реже, чем функцию ценности. Сеть алгоритмов управления остается неизменной до тех пор, пока ошибка значения не станет достаточно малой.

Сглаживание целевого алгоритма: в TD3 стратегия сглаживающей регуляризации применяется к функции стоимости, чтобы убедиться, что детерминированная методика не перестраивается под узкие пики. Мы просто добавляем небольшое количество случайных шумов к выбранным действиям, сжимаем их с некоторым коэффициентом и усредняем по мини-партиям.

$$y = r + \gamma * Q_w\left(s`, \mu_\theta\left(s`\right) + \xi\right),$$

где $\xi \sim \text{clip}\left(N\left(0, \sigma\right), -c, +c\right)$.

Информация о сети

Как и в большинстве реализаций «актор – критик», мы строим нашу сеть с двумя скрытыми слоями с 64 нейронами в каждом и гиперболическим тангенсом в качестве функции активации после каждого слоя. Все веса задаются с ортогональной инициализацией.

Акторная сеть используется для отображения состояния на действия. Сначала мы получаем нормальное распределение (μ, σ) для текущего наблюдения, а затем выбираем значения для наших действий. Мы ограничили значения наших действий максимумами {-200; 200}, поэтому в последнем слое мы использовали активацию гиперболическим тангенсом, умножив результат на константу.

Критическая сеть возвращает текущему наблюдению оценочное значение, которое не ограничено.

Начальная скорость обучения – $2{,}5e^{-4}$, параметр клипа – 0,1, коэффициент энтропии – 0,01.

■ Окружающая среда

Для наших рабочих целей мы искали подходящую обстановку, которая максимально отвечала бы всем нашим требованиям.

К сожалению, в процессе поиска симулятора мы столкнулись с рядом трудностей. Например, одни

симуляторы не могли моделировать динамику твердого тела в воде, другие не подходили для решения задач ОП, третьи были дорогими или недоступными для публичного использования. Мы провели предварительное исследование, чтобы определить подходящую среду моделирования. Руководствуясь идеей работы с чем-то достаточно простым и реалистичным, мы приняли окончательное решение в пользу комбинации Gazebo (www.gazebosim.org) с плагином Hydrodynamics, ROS Noetic (www.ros.org) и Gym (arxiv.org/abs/1606.01540).

Gazebo – это динамический 3D-симулятор с открытым исходным кодом, позволяющий точно и эффективно моделировать роботов в обоих типах условий: в сложных помещениях и на улице. Решение не предназначено специально для использования со средой Gym, а пакет ROS реализует архитектуру из трех логических слоев (на самом деле можно использовать любое количество), при этом самый нижний слой соединяет Gym API с Gazebo.

В нашем исследовании мы использовали среду, созданную на основе проекта «gncgym». Эта среда представляет собой плоский водный мир, в котором агент взаимодействует с другими объектами.

Для нашей задачи мы взяли абстрактную упрощенную модель судна с тремя степенями свободы движения. Это судно выступает в качестве агента в нашем окружении. Модель нашего судна имеет две системы управления: угол поворота руля и мощность двигателя.

Вся окружающая среда смоделирована в Gazebo и представляет собой открытый океан без каких-либо естественных препятствий. При необходимости можно добавить плавающие препятствия, такие как буи или корабли. Среда включает в себя моделирование природных сил, таких как ветер и течение. Агент интегрирован в OpenAI Gym, что особенно удобно для задач ОП.

Наша основная цель – научить ОП-агента следовать по заданному маршруту между двумя точками, учитывая препятствия, поэтому мы добавляем движущиеся и статичные объекты в зависимости от сценария и его уровня сложности.

Процесс обучения

В этом исследовании мы изучили различные сценарии, постепенно повышая уровень сложности. Базовый сценарий призван продемонстрировать способность агента выполнять простейшее следование по маршруту, в то время как последующие требуют от агента предельной ловкости действий в нетривиальных динамически меняющихся ситуациях.

В среде агент живет по эпизодам. Каждый эпизод начинается для каждого сценария с того, что агент находится в точке отправления, и его цель – прибыть в точку назначения, следуя предложенному пути между ними как можно точнее. Координаты обеих точек выбираются случайным образом в каждом эпизоде. В общем случае эпизод заканчивается, когда агент прибывает в точку назначения или если превышен лимит времени эпизода. Для каждого сценария существует свой параметр окончания эпизода.

Следование по прямой линии

Вначале мы хотели убедиться, что агент способен выполнить простую задачу – следовать по прямой. Дополнительное наказание к общему случаю, которое он получил в этом сценарии, – штраф за отклонение от маршрута. Каждый раз, когда агент сбивается с маршрута, он получает отрицательное вознаграждение, пропорциональное расстоянию. Если агент превышал пределы удаленности от маршрута, эпизод заканчивался. После того как агент достаточно хорошо справился с этой задачей, мы перешли к более сложной.

Движение по прямой линии со статичными препятствиями

В этом сценарии агенту предлагалось действовать точно так же, как и в предыдущем, но на этот раз в каждом эпизоде агент мог находить препятствия, случайным образом расположенные рядом или даже прямо на маршруте. Все препятствия были строго статичны (не меняли своего положения в течение одного эпизода). На этот раз агент получал огромный штраф в случае столкновения с любым препятствием. Факт столкновения подтверждался, если агент приближался к препятствию ближе, чем радиус круга безопасности (половина длины судна в целом). В дополнение к общим случаям окончания эпизода, эпизод заканчивается сразу после столкновения.

Движение по криволинейной траектории со статическими препятствиями

Единственное отличие от предыдущего сценария заключается в том, что здесь агент должен следовать не по прямой, а по кривой линии. В каждом эпизоде ее форма, а также координаты начальной и целевой точек задаются случайным образом. Препятствия могут появляться в разных ракурсах, их размеры также варьируются.

Движение по криволинейной траектории с динамическими препятствиями

Последний сценарий позволил нам уточнить поведение, которое мы искали. Агент должен стараться как можно точнее следовать по извилистому маршруту, избегая столкновений с любыми препятствиями, независимо от того, движутся они или нет. В данном случае строгость штрафа за отклонение от маршрута была значительно снижена, а также увеличен допустимый коридор для движения.

Заключение

Мы были вынуждены изучить различные аспекты этой технологии, чтобы создать автономный надводный корабль. В своих исследованиях мы предложили способ управления движением корабля, который позволяет перемещаться из точки в точку по определенной схеме. Наш главный вывод заключается в том, что сочетание логики следования схеме и логики обхода препятствий успешно работает и дополняет друг друга. Нам удалось сформировать агента, который может обеспечить постоянное и надежное поведение для сценария следования по извилистой траектории с динамическими препятствиями. Продемонстрировано, что такой алгоритм управления, как PPO, усваивает вполне очевидную зависимость между крутящим моментом двигателя, углом поворота руля и движением в определенном направлении. Что касается дальнейших шагов, то в будущем мы хотим усложнить модель с помощью судна и включить новую логику планирования пути агента.

■ Будущие исследования

В данной работе мы исследовали малоизвестную область алгоритмов подкрепляющего обучения с непрерывным управлением применительно к водным судам. Они показали свою эффективность для задачи управления объектом в среде с нетривиальной динамикой.

Важность соответствующего симулятора имеет решающее значение как с точки зрения реалистичного динамического моделирования, так и с точки зрения процесса обучения ОП. Мы предполагаем попробовать другой симулятор или, возможно, адаптировать наше текущее решение, чтобы полностью удовлетворить наши требования.

Мы можем предположить, что модель судна, представленная в данной работе, может быть управляемым объектом, но все же она слишком проста для тщательного исследования. В будущем будет предложена более детальная реализация реального морского судна. Судно будет оснащено носовым и кормовым азимутальными подруливающими устройствами, чтобы иметь возможность работать в режиме динамического позиционирования.

Логика избегания препятствий – важная часть автономных надводных кораблей. Хотя мы использовали в качестве препятствий только свободно плавающие объекты, в море ситуация иная, где другие корабли могут рассматриваться как агенты. Наша будущая концепция будет строиться вокруг мультиагентной системы с динамическим механизмом предотвращения столкновений.

Агент должен не только уметь перемещать корабль в произвольном направлении, но и предлагать логику планирования пути, чтобы построить маршрут между двумя точками. Мы планируем добавить еще одного агента, который будет отвечать за построение маршрута, разделенного путевыми точками на ряд секций, представляющих собой схему, которой нужно следовать.

СПИСОК ИСТОЧНИКОВ

1. EMSA Consolidated Annual Activity Report 2020, Ref. Ares (2021)3908717 – 15/06/2021
2. Figueiredo J.M.P., Rejaili R.P.A. Deep Reinforcement Learning Algorithms for Ship Navigation in Restricted Waters // Mecatrone. 2018. Vol. 3, No. 1. P. 1.
3. Sawada R., Sato K., Majima T. Automatic ship collision avoidance using deep reinforcement learning with LSTM in continuous action spaces // Journal of Marine Science and Technology. 2021. Vol. 26, No. 1. P. 509–524. https://doi.org/10.1007/s00773-020-00755-0
4. Zhu M., Skulstad R., Zhao L., Zhang H., Li G. MPC-based path planning for ship avoidance under COLREGS. 2022. URL: https://www.researchgate.net/publication/365588516_MPC-based_path_planning_for_ship_collision_avoidance_under_COLREGS
5. Imazu H., Koyama T. The optimization of the criterion for collision avoidance action // The Journal of Japan Institute of Navigation. 1985. Vol. 71. P. 123–130.
6. Kouzuki A., Hasegawa K. Automatic collision avoidance system for ships using fuzzy control // Journal of the Kansai Society of Naval Architects. 1987. Vol. 205. P. 1–10.
7. Stamenkovich M. An application of artificial neural networks for autonomous ship navigation through a channel // IEEE PLANS 92 Position Location and Navigation Symposium Record. Troy, MI, USA, 1992. P. 346–352.

8. Xie S., Chu X., Zheng M., Liu Ch. A composite learning method for multi-ship collision avoidance based on reinforcement learning and inverse control. 2020. URL: https://www.sciencedirect.com/science/article/abs/pii/S0925231220309401
9. McGookin E.W., Murray-Smith D.J., Li Y., Fossen T. I. Ship steering control system optimisation using genetic algorithms // Control Eng. Pract. 2000. Vol. 8. P. 429–443. DOI: 10.1016/S0967-0661(99)00159-8
10. Fossen T.I. Handbook of Marine Craft Hydrodynamics and Motion Control. John Wiley & Sons, 2011. 582 p.
11. Schulman J., Levine S., Moritz Ph., Jordan M.I., Abbeel P. Trust Region Policy Optimization. arXiv.org > cs > arXiv:1502.05477
12. Schulman J., Wolski F., Dhariwal P., Radford A., Klimov O. Proximal Policy Optimization Algorithms. arxiv.org/abs/1707.06347
13. Wang N., Su S.-F., Yin J., Zheng Z., Er M. J. Global asymptotic model-free trajectory-independent tracking control of an uncertain marine vehicle: an adaptive universe-based fuzzy control approach // IEEE Trans Fuzzy Syst. 2017. Vol. 26. P. 1613–1625. DOI: 10.1109/TFUZZ.2017.2737405
14. Singh A.J., Nguyen D.T., Kumar A., Lau H.C. Multiagent decision making for maritime traffic management. 2019. URL: https://ink.library.smu.edu.sg/cgi/viewcontent.cgi?article=5890&context=sis_research

Справка об авторах

ФАРХАДОВ Маис Паша оглы, д.т.н., зав. лаб. эргатических систем, г. н. с.
Институт проблем управления имени В.А. Трапезникова Российской академии наук
Адрес: 117997, г. Москва, ул. Профсоюзная, дом 65. Россия
Область научных интересов: теория, технологии гетерогенных систем и сетей обслуживания и управления, модели, алгоритмы и технические средства управления робототехническими комплексами и сетями, фундаментальные проблемы эргатических систем, многофункциональные подводные аппараты
E-mail: mais@ipu.ru.
Тел.: +7 925 507 8195
ORCID: https://orcid.org/0000-0002-7166-9567
Scopus AuthorId=57195741457
ResearcherID: S-7814-2016.
AuthorID: 205385

СОРОКИН Андрей Николаевич, соискатель
Институт проблем управления имени В. А. Трапезникова Российской академии наук
Адрес: 117997, г. Москва, ул. Профсоюзная, дом 65. Россия
Область научных интересов: Навигация и управление в подводном и надводном пространстве, машинное обучение, искусственный интеллект.
E-mail: nasir84@mail.ru

ЖИДОМИРОВ Иван Федорович, соискатель
Институт проблем управления имени В. А. Трапезникова Российской академии наук
Адрес: 117997, г. Москва, ул. Профсоюзная, дом 65. Россия
Область научных интересов: Навигация и управление в подводном и надводном пространстве, машинное обучение, радио и акустическая связь, беспилотные аппараты
E-mail: enegazer@yandex.ru
ORCID: https://orcid.org/0000-0001-6791-2774
SPIN-код: 8231-2102

КЕРИМОВ Сервер Сейранович, **инженер**
Институт проблем управления имени В. А. Трапезникова Российской академии наук
Адрес: 117997, г. Москва, ул. Профсоюзная, дом 65. Россия
Область научных интересов: управление беспилотными аппаратами, навигация и управление в подводном пространстве, информационные и управляющие модели и архитектуры.
E-mail: serverdevel@ya.ru

Для цитирования:

Фархадов М.П., Сорокин А.Н., Жидомиров И.Ф., Керимов С.С. АВТОНОМНОЕ СЛЕДОВАНИЕ БЕЗЭКИПАЖНОГО СУДНА ПО ТРАЕКТОРИИ И ПРЕДОТВРАЩЕНИЕ СТОЛКНОВЕНИЙ // Подводные исследования и робототехника. 2024. №. 3 (49). С. 52–61. DOI: 10.37102/1992-4429_2024_49_03_05. EDN: OHDAIT.

DOI: 10.37102/1992-4429_2024_49_03_05

AUTONOMOUS VESSEL PATH FOLLOWING AND COLLISION AVOIDANCE

M. Farkhadov, A. Sorokin, I. Zhidomirov, S. Kerimov

In this paper we propose a method to solve the problem of autonomous vessel navigation in highly uncertain conditions. The primary goal was to make the Reinforcement Learning agent to learn the policy allowing the AV to follow a certain path while avoid collisions with any other objects. Maneuvering of AV in the proposed conditions is the principal subject of this paper. Several scenarios were explored during the research with static and dynamic objects. The agent was trained with model free off-policy algorithms. The training process was divided in several parts where we were experimenting with meta-learning approach elements to achieve the robustness of the agent behavior.

Keywords: path following, maneuvering, machine learning, reinforcement learning, model free, obstacle avoidance.

References

1. EMSA Consolidated Annual Activity Report 2020, Ref. Ares (2021)3908717 – 15/06/2021

2. Figueiredo J.M.P., Rejaili R.P.A. Deep Reinforcement Learning Algorithms for Ship Navigation in Restricted Waters. Mecatrone. 2018. Vol. 3, No. 1. P. 1.

3. Sawada R., Sato K., Majima T. Automatic ship collision avoidance using deep reinforcement learning with LSTM in continuous action spaces. Journal of Marine Science and Technology. 2021. Vol. 26, No. 1. P. 509–524. https://doi.org/10.1007/s00773-020-00755-0

4. Zhu M., Skulstad R., Zhao L., Zhang H., Li G. MPC-based path planning for ship avoidance under COLREGS. 2022. https://www.researchgate.net/publication/365588516_MPC-based_path_planning_for_ship_collision_avoidance_under_COLREGS

5. Imazu H., Koyama T. The optimization of the criterion for collision avoidance action. The Journal of Japan Institute of Navigation. 1985. Vol. 71. P. 123–130.

6. Kouzuki A., Hasegawa K. Automatic collision avoidance system for ships using fuzzy control. Journal of the Kansai Society of Naval Architects. 1987. Vol. 205. P. 1–10.

7. Stamenkovich M. An application of artificial neural networks for autonomous ship navigation through a channel. IEEE PLANS 92 Position Location and Navigation Symposium Record. Troy, MI, USA, 1992. P. 346–352.

8. Xie S., Chu X., Zheng M., Liu Ch. A composite learning method for multi-ship collision avoidance based on reinforcement learning and inverse control. 2020. https://www.sciencedirect.com/science/article/abs/pii/S0925231220309401

9. McGookin E.W., Murray-Smith D.J., Li Y., Fossen T.I. Ship steering control system optimisation using genetic algorithms. Control Eng. Pract. 2000. Vol. 8. P. 429–443. DOI: 10.1016/S0967-0661(99)00159-8

10. Fossen T.I. Handbook of Marine Craft Hydrodynamics and Motion Control. John Wiley & Sons. 2011. 582 p.

11. Schulman J., Levine S., Moritz Ph., Jordan M.I., Abbeel P. Trust Region Policy Optimization, arXiv.org > cs > arXiv:1502.05477

12. Schulman J., Wolski F., Dhariwal P., Radford A., Klimov O. Proximal Policy Optimization Algorithms, arxiv.org/abs/1707.06347

13. Wang N., Su S.-F., Yin J., Zheng Z., Er M. J. Global asymptotic model-free trajectory-independent tracking control of an uncertain marine vehicle: an adaptive universe-based fuzzy control approach. IEEE Trans Fuzzy Syst. 2017. Vol. 26. P. 1613–1625. DOI: 10.1109/TFUZZ.2017.2737405

14. Singh A.J., Nguyen D.T., Kumar A., Lau H.C. Multiagent decision making for maritime traffic management. 2019. https://ink.library.smu.edu.sg/cgi/viewcontent.cgi?article=5890&context=sis_research

Information about the authors

FARHADOV Mais, Doctor of Technical Sciences, Head of the Laboratory of Ergatic Systems, Chief Researcher
V.A. Trapeznikov Institute of Management Problems of the Russian Academy of Sciences
Work address: 65, Trade Union Street, Moscow, 117997. Russia
Research Interests : theory, technologies of heterogeneous systems and service and control networks, models, algorithms and technical means of controlling robotic complexes and networks, fundamental problems of ergatic systems, multifunctional underwater vehicles
E-mail: mais@ipu.ru. **Phone:** +7 925 507 8195
ORCID: https://orcid.org/0000-0002-7166-9567
Scopus AuthorId=57195741457. **AuthorID**: 205385
Web of Science ResearcherID: S-7814-2016

SOROKIN Andrey, applicant
V.A. Trapeznikov Institute of Management Problems of the Russian Academy of Sciences
Work address: Moscow, Russia
Research Interests Navigation and control in underwater and surface space, machine learning, artificial intelligence.
E-mail: nasir84@mail.ru

ZHIDOMIROV Ivan, applicant
V.A. Trapeznikov Institute of Management Problems of the Russian Academy of Sciences, VTB PAO
Work address: Moscow, Russia
Research Interests Navigation and control in underwater and surface space, machine learning, radio and acoustic communication, unmanned vehicles
E-mail: enegazer@yandex.ru
ORCID: https://orcid.org/0000-0001-6791-2774

KERIMOV Server, engineer
V.A. Trapeznikov Institute of Management Problems of the Russian Academy of Sciences
Work address: Moscow, Russia
Research Interests control of unmanned vehicles, navigation and control in underwater space, information and control models and architectures.
E-mail: serverdevel@ya.ru