УДК 004.032.26; 621.396 **DOI** 10.15622/ia.25.2.2

А.А. Толстых, А.Н. Голубинский

ИНТЕЛЛЕКТУАЛЬНЫЙ АНАЛИЗ ДАННЫХ НА БАЗЕ ГЛУБОКОГО ОБУЧЕНИЯ С ПОДКРЕПЛЕНИЕМ ДЛЯ ПРОГНОЗА РАБОЧИХ ЧАСТОТ И ПОЛОС В СИСТЕМЕ КОГНИТИВНОГО РАДИО

Толстых А.А., Голубинский А.Н. **Интеллектуальный анализ данных на базе глубокого обучения с подкреплением для прогноза рабочих частот и полос в системе когнитивного радио.**

Аннотация. В работе предложен метод решения задачи выбора канала связи в когнитивном радио на основе информации о текущем состоянии всех доступных каналов связи с использованием математического аппарата обучения с подкреплением. Метод заключается в формализации задачи выбора каналов связи в терминах «среда-агент» и обучении агентов с помощью алгоритмов Reinforce, SARSA и A2C. Приведён расчёт затрат памяти на решение задачи выбора каналов связи классическими методами. Оценка по памяти составляет $\mathbf{4{\times}2^{2n}}$ байт для случайного состояния каналов (занят/свободен) и $\mathbf{4{\times}}n^2$ байт – для одного свободного канала на каждом шаге при решении задачи табличным алгоритмом Q-обучения. Приведены две различные формализации вознаграждения для агента в рамках решаемой задачи при использовании обучения с подкреплением – для тривиального случая (бинарная доступность / недоступность частотного канала) и для более сложного случая – с учётом мощности (в дБ) в выбранном канале связи. Ограничение на первую формализацию состоит в том, что на каждой итерации должен быть только один свободный канал связи из всех доступных. Вторая предложенная формализация функции вознаграждения не накладывает подобных ограничений и более универсальна. Проведены вычислительные эксперименты для обеих формализаций функции вознаграждения, агенты обучающиеся с помощью алгоритмов SARSA и A2C, в среднем, достигают безошибочного решения задачи за 8000 эпизодов обучения для обеих формализаций обучения в модельной задаче для различных реализаций агентов. Алгоритм REINFORCE не позволяет достигать безошибочного решения, однако, формализация вознаграждения с учётом мощности повышает стабильность обучения алгоритмом REINFORCE. Даны теоретические оценки вычислительной сложности рассматриваемых методов, согласующиеся с вычислительными экспериментами.

Ключевые слова: когнитивное радио, обучение с подкреплением, глубокое обучение, искусственная нейронная сеть, многослойный персептрон, функция вознаграждения, программно-определяемое радио, синтетические данные, аугментация, искусственный интеллект.

1. Введение. Функционирование когнитивных систем радиосвязи основывается на мониторинге радиочастотного спектра с целью идентификации и дальнейшего использования свободных полос частот. Классические подходы не всегда позволяют получить приемлемый на практике результат, в связи с этим, одним из возможных путей решения является применение технологий искусственного интеллекта [1 – 4], и, в частности, обучения с подкреплением [2, 5 – 14]. Таким образом, современный этап научно-технического развития

характеризуется переходом от автоматизации к интеллектуализации управления, при котором окончательное решение принимается интеллектуальной системой, на основании разнородных данных значительного объёма.

Следует отметить, что система когнитивного радио (CRS) – это радиосистема, использующая технологию, которая позволяет этой системе получать знания о своей среде эксплуатации и географической среде, об установившихся правилах и о своём внутреннем состоянии, динамически и автономно корректировать свои эксплуатационные параметры и протоколы согласно полученным знаниям для достижения заранее поставленных целей, обучаясь на основе полученных результатов [15].

Современное развитие аппаратных средств, программного обеспечения и баз данных, для эффективного нейросетевого моделирования на базе глубоких искусственных нейронных сетей, позволяет на сегодняшний день выделить актуальное для теории и практики научно-техническое направление – применение глубокого обучения с подкреплением для задач когнитивного радио [16 – 18].

Применение нейросетевого предиктора на основе глубокого обучения с подкреплением, позволяет проводить машинное обучение без разметки данных при использовании функции вознаграждения [19, 20], то есть при отсутствии «учителя».

Важным компонентом при этом является подготовка обучающих, валидационных и тестовых выборок, включающих композицию из реальных (например, сигналы, поступающие от трансивера на базе программно-определяемого радио или SDR- радиоприёмника) и аугментацию в виде синтетических данных для реализации нейросетевых предикторов (с формированием соответствующей базы данных для обучения).

Необходимо подчеркнуть, что отдельным приоритетным вопросом при решении поставленной задачи когнитивного радио является разработка и исследование адекватного критерия эффективности функционирования системы (например, состоящей из совокупности средств радиосвязи для различных режимов работы).

Цель работы – разработка метода прогноза рабочих частот и полос для системы когнитивного радио на базе глубокого обучения с подкреплением.

2. Постановка задачи. Рассмотрим радиоканал, в котором рабочий частотный диапазон разбит на полосы, и каждая полоса может быть занята или свободна для радиообмена данными. Математически это можно формализовать, сравнивая текущее значение мощности

(PdB_k) помехи в полосе с некоторым порогом (β), получив бинарное состояние: «1» – частотный канал (k) свободен, «0» – частотный канал занят:

$$\mathrm{PN}_k = \begin{cases} 1, \text{если } \mathrm{PdB}_k \le \beta; \\ 0, \text{иначе,} \end{cases} \quad (1)$$

где порог выбирается из состояния помехово-шумовой обстановки (например, β = -80дБм).

Требуется определить рабочие частоты и наибольшие полосы частот для динамически меняющейся электромагнитной обстановки (например, в которой присутствуют периодические и непериодические помехи, фон и другие шумы различной природы).

3. Описание метода. Определим в качестве критерия для решения поставленной задачи – максимизацию целевой функции в виде некоторого показателя качества прогноза. В этой связи необходимо максимизировать качество прогноза рекомендуемых свободных каналов:

$$\max[g_t], \quad (2)$$

где g_t – показатель качества прогноза, который определяется как функция от состояния радиоканалов в следующий момент времени. Например, как скалярное произведение соответствующих векторов и выглядит следующим образом:

$$g_t = a_t^T \mathrm{PN}_{t+1}, \quad (3)$$

здесь a_t – вектор-столбец действия агента в текущий момент времени («1» – канал рекомендуется использовать, «0» – канал не рекомендуется использовать); PN_{t+1} – вектор-столбец состояния радиоканалов на основе измеренной или моделируемой помехово-шумовой обстановки в следующий момент времени с элементами $\{\mathrm{PN}_k\}$ («1» – канал свободен, «0» – канал занят); «T» – операция транспонирования.

Функцию вознаграждения (r_{t+1}) можно определить, через функциональную зависимость от показателя качества прогноза или в первом приближении, возможно, использовать их равенство:

$$r_{t+1} = g_t. \quad (4)$$

Два основных подхода к обучению с подкреплением для безмодельных методов (не задействуют динамику переходов среды в явном виде) – алгоритмы, основанные на полезности и алгоритмы на базе стратегии (политики) [19, 20].

Например, если для решения задачи воспользоваться Q-обучением (основано на полезности), то на каждом шаге на основе ε-жадной стратегии, текущего состояния среды (s_t) и Q-матрицы (Q_t) формируется текущее действие агента a_t, на базе которого получается состояние среды, т.е. матрица прогноза состояния радиоканала («1» – канал свободен, «0» – канал занят) в следующий момент времени s_{t+1}, например:

$$s_{t+1} = a_t, \tag{5}$$

которая также может быть дополнена априорной информацией, помимо вектора-столбца a_t, векторами-столбцами о текущем и предыдущих состояниях радиоканала (например: PN_{t+1}, PN_t, PN_{t-1},…). Далее рассчитывается значение функции вознаграждения в следующий момент времени (r_{t+1}) и на его основе вычисляется Q-матрица значений ценности состояний в следующий момент времени (Q_{t+1}), используя такие гиперпараметры как скорость обучения (α) и коэффициент дисконтирования (γ) [19]. Настройка Q-функции осуществляется с помощью метода временных различий (TD-обучения, сочетающего в себе идеи метода Монте-Карло и динамического программирования) и использования итерационного метода для решения уравнения оптимальности Беллмана [20].

Существенным ограничением для непосредственного использования алгоритмов табличных методов решения на основе полезности и на основе стратегии является значительное множество возможных состояний (для Q-обучения – это большая размерность Q-матрицы, а для обучения на базе стратегии – большая размерность π-матрицы вероятностей действия в соответствующем состоянии). Например, при n одновременно анализируемых частотных каналов при случайном состоянии канала («свободен»/«занят») количество состояний среды (N) определяется выражением (количество строк Q-матрицы):

$$N = 2^n, \tag{6}$$

а количество возможных действий агента (M) также рассчитывается по формуле (количество столбцов Q-матрицы):

$$M = 2^n. \quad (7)$$

В результате Q-матрица содержит N×M элементов:

$$N \times M = 2^{2n}, \quad (8)$$

и для её описания при одинарной точности (FP32) потребуется:

$$V = 4 \times 2^{2n} \text{ байт}. \quad (9)$$

При n=16 следует, что только для хранения Q-матрицы потребуется 16 ГБ, а при n=20 необходимо 4 ТБ.

Однако, если рассмотреть тривиальный частный случай, например, когда из n каналов всегда свободен только один, то размер Q-матрицы существенно уменьшается, а расчётные формулы принимают вид:

$$N = n; \quad M = n, \quad N \times M = n \times n, \quad V = 4\,n^2 \text{ байт}, \quad (10)$$

тогда при n=16 следует, что для хранения Q-матрицы потребуется 1 КБ, а при n=20 необходимо 1,6 КБ.

Таким образом, наличие закономерностей (например, детерминированных) появления помех в радиоканалах позволяет определённым образом уменьшить число возможных состояний, однако следует учитывать условия конкретной задачи и ограничения, накладываемые на объем памяти ОЗУ. Так как для агента очень важно, какие вычислительные мощности ему доступны, в частности какой объем вычислений может быть выполнен за один временно́й шаг, также сдерживающим фактором является доступная память [19].

Для преодоления проблемы размерности, связанной со значительным числом возможных состояний, предлагается воспользоваться компактной (относительно таблиц) параметризацией (приближенным методом решения) в виде аппроксимаций Q-функции полезности (рисунок 1) или π-функции стратегии (рисунок 2) искусственной нейронной сетью (ИНС) в виде многослойного персептрона (МСП) с параметрами w. Под глубокой нейронной сетью будем понимать ИНС, которая содержит три и более слоёв (два и более скрытых слоёв).

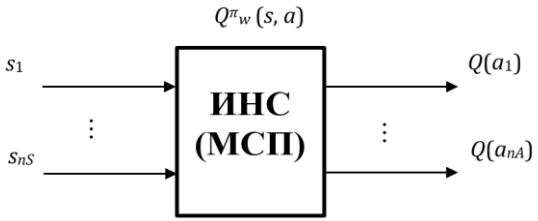

Рис. 1. Блок-схема аппроксиматора Q-функции полезности ИНС в виде МСП

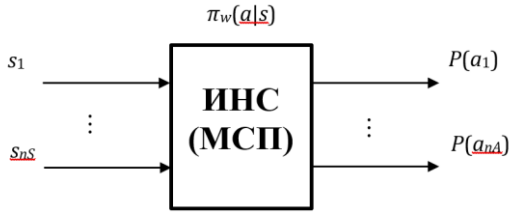

Рис. 2. Блок-схема аппроксиматора π-функции стратегии ИНС в виде МСП

При этом для глубокого обучения с подкреплением существует множество эффективных алгоритмов, основанных на полезности (SARSA, DQN), стратегии (Policy Gradients, REINFORSE), комбинированные методы (A2C, A3C) и др. [20]. Заметим, что для сокращения размерности выходного слоя ИНС (количества выходов МСП) целесообразно использовать подход, базирующийся на обучении с подкреплением при многоцелевых действиях агента [21].

Полученные значения действий агента (a_t) позволяют прогнозировать свободные (пригодные для радиосвязи) частотные каналы без разметки данных для обучения. Следует отметить, что если прогнозные значения находящихся рядом каналов «свободны» (равны «1»), то это позволяет выбрать соответствующую более широкую полосу частот на рабочей частоте (которая является центральной частотой в окрестности полосы).

3. Теоретическая оценка вычислительной сложности рассматриваемых алгоритмов. В работе рассматриваются алгоритмы следующие алгоритмы обучения с подкреплением: REINFORCE, SARSA, A2C. Перед проведением вычислительного эксперимента целесообразно оценить вычислительную эффективность каждого из них, а затем сопоставить теоретические оценки с полученным эмпирическим материалом. Формальное построение оценки

вычислительной эффективности произведено с учётом следующих ограничений:

1. Предполагается дискретное пространство действий (выбор одного из n каналов);
2. Агент в виде МСП содержит один скрытый слой;
3. Последнее выбранное действие (отклик) агента интегрировано в тензор состояния среды;
4. Вычисление функции активации будем считать не влияющий на асимптотическую сложность ($f(x) \overset{\text{Def}}{=} O(1)$).

Для построения оценки алгоритмов введём следующие обозначения:

1. Среда характеризуется размерностью состояния d_s; количеством дискретных действий d_a; макисмальной длинной эпизода T.
2. Агент (искусственная нейронная сеть) количеством нейронов в скрытом слое m; входной размерностью d_s; количеством выходных нейронов d_a.

Рассмотрим по порядку каждый из алгоритмов. Алгоритм REINFORCE представляет собой реализацию метода градиента политики (Policy Gradient), где агент обучает параметризованную политику $\pi_\theta(\mathrm{a}|\mathrm{s})$, реализованную как МСП путём максимизации ожидаемой награды. Политика генерирует действия и агрегируются траектории в рамках одного эпизода, вычисляются функция стоимости и градиент политики обновляется. Формула градиента:

$$\nabla_\theta J(\theta) = \mathbb{E}\tau \sim \pi\theta \left[\sum_{t=0}^{T} \nabla_\theta \, log\pi_\theta(a_t|s_t) \cdot G_t \right], \tag{11}$$

где $G_t = \sum_{k=t}^{T} \gamma^{k-t} r_k$ дисконтированное вознаграждение (reward), γ – дисконтирующий фактор, $\tau = (s_0, a_0, r_0, \dots, s_T)$ – траектория в рамках эпизода. Для каждого эпизода вычисляется: прямое вычисление МСП для выбора действия $O(T \cdot m^2)$, вычисление функции вознаграждения $O(T)$ и вычисление градиентов для МСП $O(T \cdot m^2)$. Таким образом, основная вычисления приходятся на прямой проход и обратное распространение ошибки в МСП. Сложность алгоритма REINFORCE линейна по T, однако в значительной мере зависит от размера МСП. Для наших целей, можно считать, что вычисление REINFORCE заключается в вычислении прямого прохода МСП и обратного распространения ошибки по нему для каждой итерации эпизода.

Алгоритм SARSA представляет собой метод аппроксимации Q-функции $Q_\theta(s,a)$ реализованной в виде МСП. Агент выбирает действия по ε-жадной стратегии. Обновление на каждом шаге определяется следующим образом [20]:

$$Q(s_t,a_t) = Q(s_t,a_t) + \alpha\big[r_t + \gamma Q\big(s_{\{t+1\}},a_{\{t+1\}}\big) - Q(s_t,a_t)\big], \quad (12)$$

где $a_{t+1} \sim \pi(s_{t+1})$ представляет собой политику; α – скорость обучения. Градиенты для МСП в алгоритме SARSA вычисляются как [20]:

$$\theta = \theta - \alpha\nabla_\theta\big(r_t + \gamma Q_\theta(s_{t+1},a_{t+1}) - Q_\theta(s_t,a_t)\big)^2. \quad (13)$$

Аналогично алгоритму REINFORCE на первом этапе вычисляется прямой проход МСП $O(T\cdot m^2)$, после чего происходит шаг симуляции $O(T)$, затем второе вычисление прямого прохода по МСП для вычисления a_{t+1} $O(T\cdot \mathrm{m}^2)$, вычисление ошибки на основе функции вознаграждения $O(T)$ и вычисление градиентов для МСП $O(T\cdot \mathrm{m}^2)$. Таким образом, асимптотически алгоритм SARSA имеет какую же сложность что и REINFORCE, однако, как и в случае с REINFORCE можно дать оценку в количестве прямых проходов и обратного распространения ошибки, для SARSA соответственно 2 и 1 на каждую итерацию в эпизоде.

Последний рассматриваемый алгоритм A2C представляет собой синхронный вариант метода актор-критик и имеет 2 МСП: актора для аппроксимации политики $\pi_\theta(\mathrm{a}|\mathrm{s})$ и критика для оценки функции стоимости $V_\varphi(s)$. Также как и в предыдущих алгоритмах, на первом этапе вычисляется прямой проход по МСП-актору $O(T\cdot \mathrm{m}^2)$, затем вычисляется прямой проход по МСП-критику $O(T\cdot \mathrm{m}^2)$, после чего вычисляются шаг симуляции и функция потерь $O(T)$, завершается все вычислением градиентов для МСП-актора $O(T\cdot \mathrm{m}^2)$ и МСП-критика $O(T\cdot \mathrm{m}^2)$. Таким образом, для алгоритма A2C сложность линейна по длине эпизода, однако на каждую итерацию необходимо вычислить 2 прямых прохода по МСП и 2 прохода обратного распространения ошибки.

Учитывая тот факт, что наибольший вклад в оценку вычислительной сложности рассматриваемых алгоритмов вносит прямой проход и обратное распространение ошибки в МСП целесообразно расширить оценку на произвольный МСП с L скрытыми слоями по $m \in \{m_1, m_2, \dots, m_L\}$ нейронов в каждом, таким образом сняв

ограничение II. Учитывая ограничение IV прямой проход по l слою L- слойного МСП оценивается как:

$$O\left(m_{\{l-1\}} \times m_{\{l\}} + m_{\{l\}}\right) \to O\left(m_{\{l-1\}} \times m_{\{l\}}\right) \xrightarrow[m=max(m_{l-1},m_l)]{} O(m^2). \quad (14)$$

Заметим, что сложность прямого прохода в терминах O равна сложности обратного распространения ошибки. В качестве верхней оценки вычислительной сложности для L-слойного МСП целесообразно принять значение $O(L \times \max(m)^2)$ – сложность вычисления каждого слоя соответствует сложности вычисления наиболее широкого (содержащего наибольшее число нейронов) слоя МСП. В рамках рассматриваемой задачи данное огрубление оценки допустимо, так как, сравниваются алгоритмы, обучающие одинаковые L- слойные МСП (в том числе МСП-критик для алгоритма SARSA имеет ту же архитектуру, что и МСП для остальных алгоритмов). Таким образом, для всех рассмотренных алгоритмов сложность остаётся линейной по T, оценка не изменяется при замене реализации аппроксиматора с однослойного персептрона на МСП.

В результате проведённого анализа, учитывая, что проход обратного распространения ошибки по МСП содержит, в среднем в 2 раза больше операций [4], сформулируем следующие количественные оценки: наиболее быстрым является алгоритм REINFORCE, приблизительно в 1,5 раза медленнее SARSA, а A2C приблизительно в 2 раза медленнее.

4. Моделирование среды. Для моделирования работы алгоритмов обучения с подкреплением на основе формализации задачи была построена следующая модель среды: в наборе из N каналов в каждый момент времени t свободен только один канал. Формально симуляция среды представляет собой правило, по которому для каждого шага времени t строится вектор из N величин соответствующих уровню шумов, индекс элемента вектора соответствует индексу канала. Среда является параметрической и определяется следующим набором параметров: N – общее число каналов, $f_{\text{пор}}(t)$ – закон изменения свободного канала; $P_{(\text{ш},min)}$ – минимальный уровень шумов (дБ); $P_{(\text{ш},max)}$ – максимальный уровень шумов (дБ); $P_{(c,min)}$ – минимальный уровень полезного сигнала (дБ); $P_{(c,max)}$ – максимальный уровень полезного сигнала (дБ); V – максимальная амплитуда флуктуаций шумов; T – общее время (дискретное) генерации, сколько векторов будет сгенерировано до окончания текущей сессии.

Подобная формализация позволяет достаточно гибко моделировать сложные ситуации, например при $P_{(c,min)} < P_{(\text{ш},max)}$ в некоторых моментах времени t полезный сигнал может оказаться ниже уровня шумов и т.д. На рисунке 3 приведена визуализация для среды с периодическим законом $f_{\text{пор}}$ для числа каналов $N = 3$ и $N = 15$.

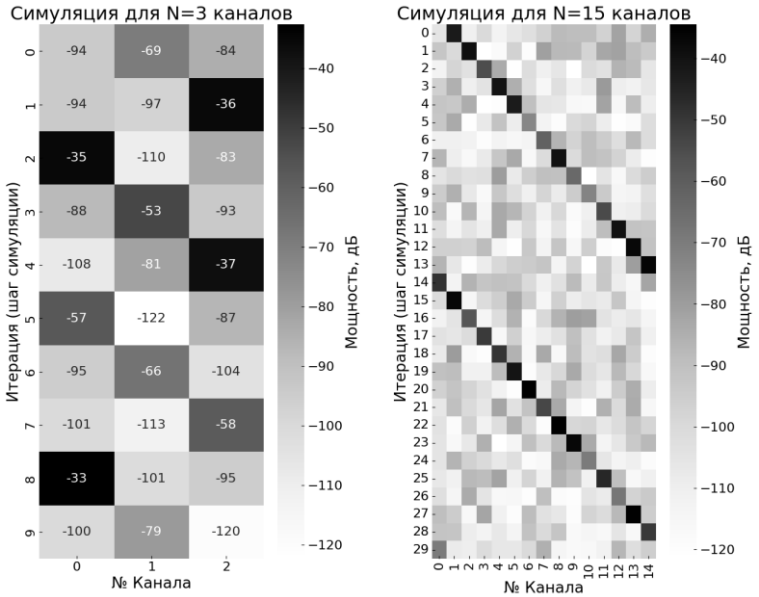

Рис. 3. Визуализация среды для $N = 3$ и $N = 15$ каналов

Порождающий закон является детерминированным и периодическим (для обеспечения генерации любого наперёд заданного количества шагов генерации T), с этой точки зрения среда является детерминированной. С другой стороны, выбор конкретных числовых значений на каждом шаге происходит по случайному закону. Рассмотрим формирование вектора состояния среды X^t на шаге t подробно. При инициализации среды для каждого канала выбирается значение из равномерного распределения $X^0 = \{x_i = U[P_{(\text{ш},min)}, P_{(\text{ш},max)}] \forall i \in N\}$ – устанавливается «фон». Далее, определяется индекс свободного канала на шаге $t = 0$ из порождающего закона $n = f_{\text{пор}}(0)$, значение $X_n^0 = U[P_{(c,min)}, P_{(c,max)}]$. Далее рассмотрим переход $t_1 \to t$: для фонового шума вычисляется изменение $X^t = \left\{x_i^{(t-1)} = x_i^{(t-1)} + U[-V, V] \forall i \in N\right\}$,

если рассчитанная величина x_i^t превышает $P_{(\text{ш},max)}$ или ниже $P_{(\text{ш},min)}$ то знак выражения меняется $\left(x_i^{(t-1)} = x_i^{(t-1)} - U[-V,V]\right)$. Таким образом, обеспечивается плавное изменение фоновых шумов. Для свободного канала операция $t_1 \to t$ аналогичная инициализации.

Технически для того, чтобы обеспечить плавное изменение уровня шума в программном классе генерации хранится вектор состояния фона (без свободного канала) и полный вектор среды раздельно. Однако такие накладные расходы в современных компьютерах пренебрежительно малы.

Предложенная среда позволяет генерировать достаточно сложные последовательности состояний, например, $f_{\text{пор}}$ может быть непериодическим или случайным. В настоящей работе был рассмотрен случай периодического порождающего закона, обобщение результатов на более широкий класс порождающих законов является предметом дальнейших работ.

5. Обсуждение результатов. В качестве примера рассмотрим среду, в которой два из $N = 3$ радиоканалов заняты, а один свободен. Положим, что номер свободного канала изменяется по детерминированному закону с периодически повторяющейся последовательностью на 30 временных интервалах. Состояние радиоканала моделируется следующими уровнями: минимальный уровень шумов -120 дБ, максимальный уровень шумов -80 дБ, минимальный уровень полезного сигнала -80 дБ, максимальный уровень полезного сигнала -120 дБ. Амплитуда флуктуации шумов 3 дБ, порождающий закон $f_{(\text{пор})}(t) = mod(t, N)$ – переодическое смещение «вправо» по каналам (по модулю N).

В качестве аппроксимации политики используется следующий набор МСП: с одним скрытым слоем, содержащим 10 нейронов; с двумя скрытыми слоями по 10 нейронов в каждом; с тремя скрытыми слоями по 10 нейронов в каждом.

Для каждого МСП из набора проводилась серия экспериментов по обучению алгоритмами REINFORCE, SARSA и A2C.

В рассмотренном для численного моделирования примере награда агента выбиралась двумя способами – в первом r_{t+1} зависит от состояния в следующий наблюдаемый момент времени, следующим образом:

$$r_{t+1} = \begin{cases} 1, & \text{PdB}_k \le -80; \\ -\left(\dfrac{\text{PdB}_k}{50} + 1{,}6\right), & \text{PdB}_k > -80, \end{cases} \tag{15}$$

где PdB_k – мощность (дБ) в выбранном (предсказанном) канале связи. Во втором – на основе выражения (4).

Для первого способа вознаграждения на основе выражения (14) на рисунке 4 представлен график для метода градиента стратегии (политики) по алгоритму REINFORSE зависимости суммарных (полных) вознаграждений и скользящее среднее по оценкам в 200 контрольных точках. Следует отметить, что максимальное полное вознаграждение составляет 30.

Оптимизация производилась с помощью метода Adam [22] со скорость обучения 0,001; параметры методов обучения с подкреплением: начальное значение $\varepsilon = 0{,}5$; конечное значение $\varepsilon = 0{,}001$; коэффициент дисконтирования $\gamma = 0{,}7$. На рисунках 4-9 используется сокращение «СВ» – суммарное вознаграждение, цифра после МСП («МСП-1») обозначает количество скрытых слоёв в МСП.

В ходе экспериментов алгоритм REINFORCE демонстрирует способность достигать оптимального (безошибочного) решения в диапазоне от 10000-12000 эпизодов обучения. Однако стоит отметить, что траектория функции вознаграждения характеризуется заметными осцилляциями, особенно на поздних этапах обучения.

На рисунке 5 представлен график для метода градиента полезности (ценности) по алгоритму SARSA зависимости суммарных (полных) вознаграждений и соответствующее скользящее среднее для способа вознаграждения на основе выражения (15).

Алгоритм SARSA, демонстрирует достижение оптимального (безошибочного) решения в районе 8000 эпизодов обучения. Такая относительно высокая скорость сходимости по сравнению REINFORCE, может быть объяснена фундаментальными особенностями SARSA: механизмом самонастройки (bootstrapping), где обновление Q-значений для текущей пары (состояние, действие) зависит от оценки следующего действия, выбранного в соответствии с той же политикой. В частности, правило обновления SARSA подразумевает, что агент оценивает текущее действие на основе предсказанной ценности следующего состояния и действия $Q(s', a')$, где a' выбирается не оптимально (как в Q-Learning), а по текущей, возможно, случайной, политике – например, на ранних этапах ε-жадной стратегии выбора действий. С другой стороны, после снижения ε до практически нулевых значений (около 8000 эпизодов) дополнительная оценка ценности следующего действия приводит к более быстрой сходимости к безошибочному решению и снижению осцилляций графика суммарного вознаграждения (которые обусловлены ненулевым значением ε на протяжении всего обучения).

Аналогичной особенностью обусловлена и повышенная дисперсия функции суммарного вознаграждения, что представлено на рисунке 4: поскольку обновления опираются на коррелированные образцы из траекторий, генерируемых текущей политикой, случайные колебания в выборе действий (включая случайные шаги на начальных итераций ε-жадной стратегии выбора действий) усиливают шум в оценках, приводя к более высокой амплитуде колебаний функции.

На рисунке 6 представлен график для комбинированного метода (актор-критика с преимуществом) по алгоритму A2C зависимости суммарных (полных) вознаграждений и соответствующее скользящее среднее для способа вознаграждения на основе выражения (14).

Алгоритм A2C), демонстрирует достижение оптимального (безошибочного) решения около 8000 эпизодов обучения. Скорость сходимости аналогична SARSA, однако порождена другим механизмом. Она, в первую очередь, обусловлена использованием двух нейронных сетей: МСП-актора для аппроксимации политики $\pi(a|s)$ и МСП-критика для оценки функции ценности $V(s)$, которая прогнозирует ожидаемое дисконтированное суммарное вознаграждение по текущему состоянию. На основе этой оценки производится более эффективная оптимизация по сравнению с алгоритмами, строящих оценки на основе метода Монте-Карло. Однако, как было показано в разделе 4, алгоритм A2C в 2 раза медленнее REINFOCE, следовательно, быстрая сходимость по эпизодам компенсируется высокой вычислительной сложностью каждой эпизода. Этим же объясняется наличии больших (по амплитуде и частоте) осцилляций графика суммарного вознаграждения по сравнению с SARSA, так как МСП-критик даже на поздних эпохах обучения может производить неверные прогнозы функции ценности $V(s)$.

Анализ графиков, представленных на рисунках 3-5, позволяет сделать вывод о том, что алгоритмы SARSA и A2C достигают полного (безошибочного) предсказания свободного канала в модельной среде за одинаковое (в среднем) количество эпизодов. Алгоритм REINFORCE не достигает уровня полного (безошибочного) предсказания. Следует отметить, что графики скользящего среднего получены по серии экспериментов (по 10 для каждого алгоритма и каждой реализации МСП) и отражают общий характер, а не конкретную реализацию, траектория обучения которой достаточно сильно зависит от начальной инициализации весов МСП.

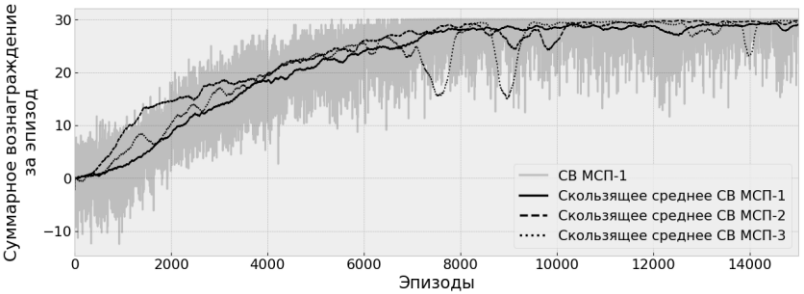

Рис. 4. Полное вознаграждение и скользящее среднее для алгоритма REINFORSE

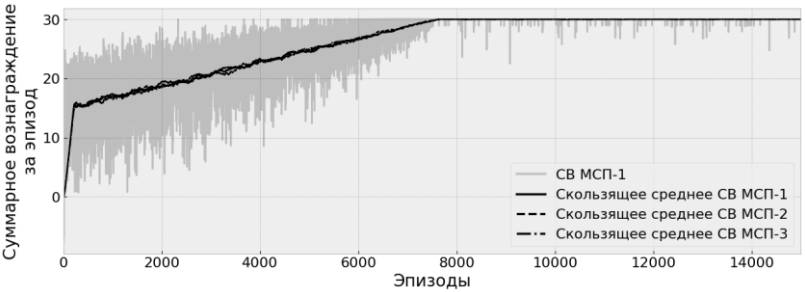

Рис. 5. Полное вознаграждение и скользящее среднее для алгоритма SARSA

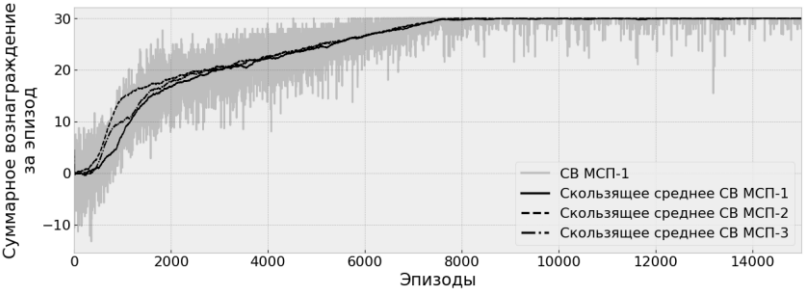

Рис. 6. Полное вознаграждение и скользящее среднее для алгоритма A2C

Для второго способа вознаграждения на основе выражения (4) на рисунках 7-9 представлены графики соответственно для алгоритмов REINFORSE, SARSA и A2C зависимости суммарных (полных) вознаграждений и скользящее среднее.

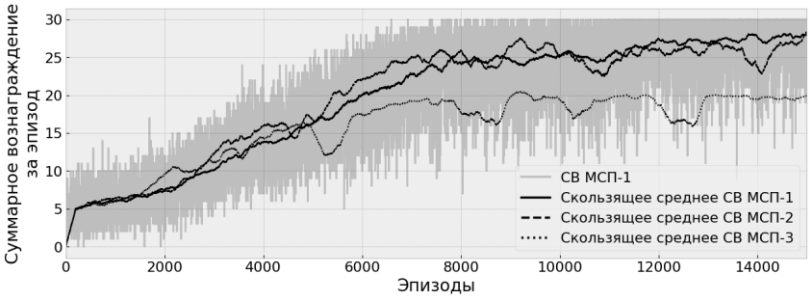

Рис. 7. Полное вознаграждение и скользящее среднее для алгоритма REINFORSE

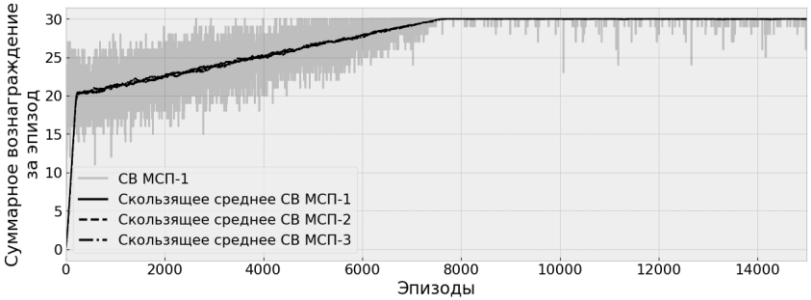

Рис. 8. Полное вознаграждение и скользящее среднее для алгоритма SARSA

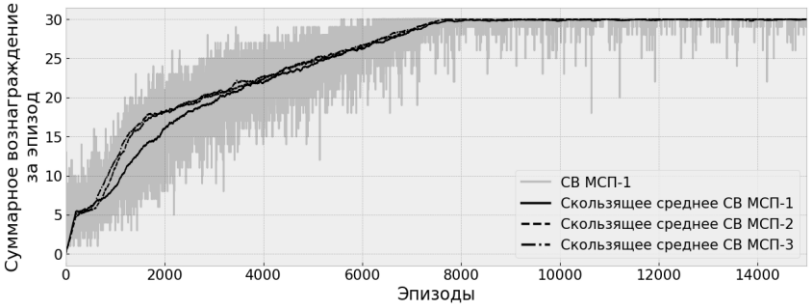

Рис. 9. Полное вознаграждение и скользящее среднее для алгоритма A2C

Из сравнения графиков на рисунках 7-9 и рисунках 4-6 следует, что выбор конкретной формализации функции вознаграждения (4) или (15) не оказывает существенного влияния на общую сходимость рассмотренных алгоритмов SARSA и A2C, однако функция (15) значительно улучшает сходимость алгоритма REINFORCE.

На рисунке 10 приведены графики среднего времени вычисления одного эпизода в разрезе алгоритмов и реализаций МСП. Следует отметить, что все эксперименты проводились в одинаковых условиях (зафиксирована программная реализация библиотеки вычисления градиентов, аппаратная составляющая и программная реализация моделируемой среды).

На рисунке 10 высота столбца соответствует среднему времени вычисления одного эпизода обучения; черные линии – стандартное отклонение 3σ (99,7% эпизодов). Из рисунка 9 следует, что оценки приведённые в разделе 4 подтверждаются вычислительным экспериментом: алгоритм REINFORCE является наиболее быстрым, SARSA медленнее (в среднем) в 1,67 раз, A2C медленнее относительно REINFORCE (в среднем) в 2,10 раз.

Другим важным выводом является тот факт, что формализации функции вознаграждения (4) или (15) не оказывает значительного влияния на сходимость алгоритмов SARSA и A2C, однако формализация (15) является более универсальной и повышает эффективность алгоритма REINFORCE. Таким образом, формализация (15) может быть рекомендована как отправная точка в решении практических задач выбора свободного канала.

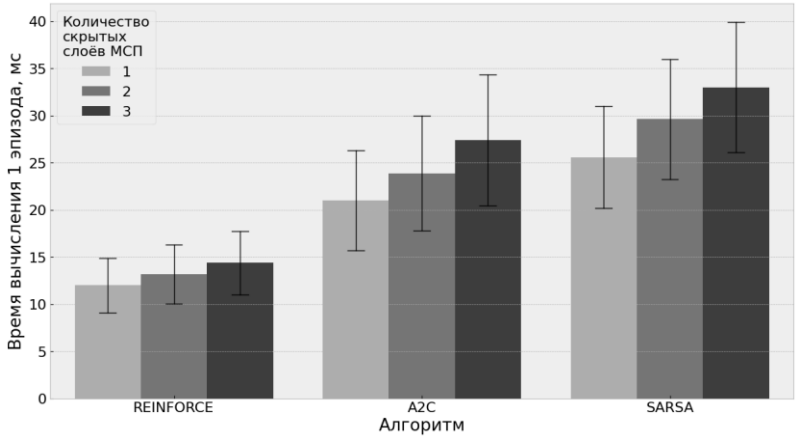

Рис. 10. Зависимость среднего суммарного вознаграждения от времени

На рисунке 11 приведён сводный график для обучения агентов алгоритмом REINFORECE для $N \in [3{,}100]$. Все внутренние параметры среды и агентов фиксированы (численные значения совпадают со значениями, приведёнными в начале раздела 5 для $N = 3$). Для

каждого N выбирается число временных интервалов как $T = N \times 10$, таким образом, для детально рассмотренного случая $N = 3: T = 30$, для $N = 100: T = 1000$ и т.д.

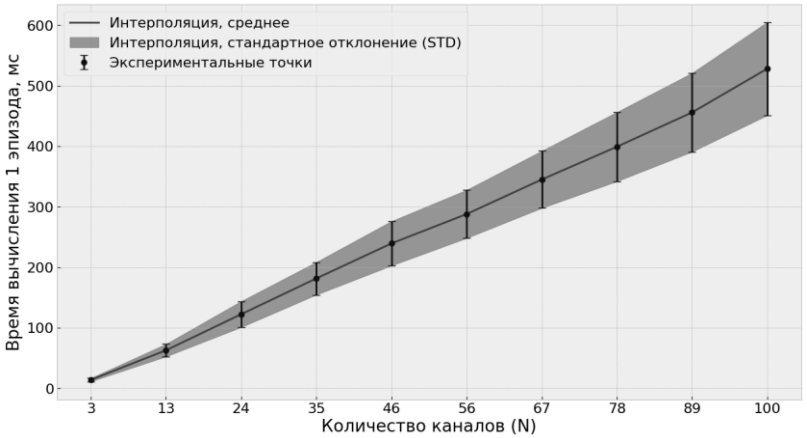

Рис. 11. Сводный график обучения агентов для $N \in [3{,}100]$

Следует отметить, что время растёт приблизительно как $O(N)$, однако, подобная тенденция должна быть исследована в областях больших ($N > 10^3$), так как вероятно, увеличение сложности изменит своё поведение. Подобное исследование целесообразно при практической заинтересованности в агентах, работающих с тысячами каналов. С другой стороны, для рассматриваемой модели среды обеспечивается устойчивое обучение агента до безошибочного решения в рассмотренном диапазоне каналов. Таким образом, можно сделать вывод о достаточно эффективном обобщении решения задачи на большее число каналов, при учёте сохранения параметров модели среды.

6. Заключение. Таким образом, для решения задачи формирования прогноза рабочих частот и полос частотной области предложен метод для системы когнитивного радио, базирующийся на глубоком обучении с подкреплением. Предложенный конструктивный метод позволяет оценить рабочие частоты и соответствующую ширину полосы частот при заданных условиях и ограничениях. Учитывая теоретические оценки вычислительной сложности, полученные в разделе 4 и анализируя отношения времени сходимости алгоритмов до безошибочного решения можно сделать вывод о согласованности теоретических и экспериментальных

результатов. Целью дальнейших исследований целесообразно определить анализ оценки эффективности включения дополнительной априорной информации в матрицу состояний среды и выбор функций вознаграждения для решения конкретных научно-прикладных задач когнитивного радио.

Литература

1. Тонг В., Чжу П. Сети 6G. Путь от 5G к 6G глазами разработчиков. От подключенных людей и вещей к подключенному интеллекту // М.: ДМК Пресс. 2022. 624 с.
2. Fette B.-A. Cognitive Radio Technology // Elsevier. 2006. 622 p.
3. Комашинский В.И., Смирнов Д.А. Нейронные сети и их применение в системах управления и связи // М.: Горячая линия–Телеком. 2003. 93 с.
4. Голубинский А.Н., Толстых А.А. Гибридный метод обучения сверточных нейронных сетей // Информатика и автоматизация. 2021. Т. 20. № 2. С. 463–490. DOI: 10.15622/ia.2021.20.2.8.
5. Wu C., Chowdhury K.-R., Di Felice M., Meleis M. Spectrum Management of Cognitive Radio Using Multi-Agent Reinforcement Learning // 9th International Conference on Autonomous Agents and Multiagent Systems. 2010. vol. 1–3. pp. 1705–1712. DOI: 10.1145/1838194.1838199.
6. Kiran U., Kumar P.-D., Reddy R.-K., Ranjith M. Efficient Exploration for Reinforcement Learning Based Distributed Spectrum Sharing in Cognitive Radio System // International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering. 2013. vol. 2. no. 11. pp. 5596–5604.
7. Yau K.-L.-A., Poh G.-S., Chien S.-F., Al-Rawi1 H.-A.-A. Application of Reinforcement Learning in Cognitive Radio Networks: Models and Algorithms // The Scientific World Journal. 2014. vol. 1. pp. 1–23. DOI: 10.1155/2014/209810.
8. Abolarinwa J.-A., Latiff A.-N.-M. Channel Decision in Cognitive Radio Enabled Sensor Networks A Reinforcement Learning Approach // International Journal of Engineering and Technology (IJET). 2015. vol. 7. no. 4. pp. 1394–1404.
9. Raj V., Dias I., Tholeti T., Kalyani S. Spectrum Access In Cognitive Radio Using A Two Stage Reinforcement Learning Approach // IEEE. 2018. vol. 12. no. 1. pp. 20–34. DOI: 10.1109/JSTSP.2018.2798920.
10. Tubachi S., Venkatesan M., Kulkarni A.-V., et al. Predictive learning model for Cognitive Radio using Reinforcement Learning // IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI). 2017. pp. 564–567. DOI: 10.1109/ICPCSI.2017.8391775.
11. Jang S.-J., Han C.-H., Lee K.-E., et al. Reinforcement learning-based dynamic band and channel selection in cognitive radio ad-hoc networks // J Wireless Com Network. 2019. vol. 2019. pp. 1–25. DOI: 10.1186/s13638-019-1433-1.
12. Singhal C., Thanikaiselvan V. Cross Layering Using Reinforcement Learning In Cognitive Radio-Based Industrial Internet Of Ad-Hoc // International Journal of Computer Networks & Communications (IJCNC). 2022. vol. 14. no. 4. pp. 1–17. DOI: 10.5121/ijcnc.2022.14401.
13. Talekar S., Banait S., Patil M. Improved Q-Reinforcement Learning Based Optimal Channel Selection In Cognitive Radio Networks // International Journal of Computer Networks & Communications (IJCNC). 2023. vol. 15. no. 3. pp. 1–14. DOI: 10.5121/ijcnc.2023.15301.
14. Rosen D., Rochez I., McIrvin C., Lee J., D'Alessandro K., Wiecek M., et al. RFRL Gym_A Reinforcement Learning Testbed for Cognitive Radio Applications //

International Conference on Machine Learning and Applications (ICMLA). 2023. pp. 279–286. DOI: 10.1109/ICMLA58977.2023.00046.
15. Отчет (Сектора связи Международного союза электросвязи) МСЭ-R SM.2152 (09/2009) «Определения системы радиосвязи с программируемыми параметрами (SDR) и системы когнитивного радио (CRS)».
16. Chitnavis S., Kwasinski A. Cross Layer Routing in Cognitive Radio Network Using Deep Reinforcement Learning // IEEE Wireless Communications and Networking Conference (WCNC). 2019. pp. 1–13. DOI: 10.1109/WCNC.2019.8885918.
17. Obite F., Usman A.-D., Okafor E. An overview of deep reinforcement learning for spectrum sensing in cognitive radio networks // Digital Signal Processing. 2021. vol. 113. pp. 1–18. DOI: 10.1016/j.dsp.2021.103014.
18. Tondwalkar A., Kwasinski A. Deep Reinforcement Learning for Distributed and Uncoordinated Cognitive Radios Resource Allocation. 2022. pp. 1–13. arXiv: 2205.13944v1.
19. Саттон Р. С., Барто Э. Дж. Обучение с подкреплением // М.: ДМК Пресс. 2020. 552 с.
20. Грессер Л., Кенг В.Л. Глубокое обучение с подкреплением: теория и практика на языке Python // СПб.: Питер. 2022. 416 с.
21. Wang H., Yu Y. Exploring Multi-Action Relationship in Reinforcement Learning // Springer, Cham. 2016. pp. 1–13. DOI: 10.1007/978-3-319-42911-3_48.
22. Kingma D., Ba J. Adam: A Method for Stochastic Optimization. // CoRR. 2014. T. abs/1412.6980.

Толстых Андрей Андреевич — канд. техн. наук, инженер-программист, ООО «РТК». Область научных интересов: искусственные нейронные сети, машинное обучение, обучение с подкреплением. Число научных публикаций — 65. tolstykh.aa@yandex.ru; проспект Высоковольтный, 1, 127566, Москва, Россия.

Голубинский Андрей Николаевич — д-р техн. наук, доцент, начальник отдела, Российский научный фонд (РНФ). Область научных интересов: машинное обучение, нейросетевое моделирование, автоматизированные системы управления с элементами искусственного интеллекта, обработка речевых сигналов. Число научных публикаций — 250. annikgol@mail.ru; улица Солянка, 14, 109240, Москва, Россия; р.т.: +7 (910) 346-6537.

DOI 10.15622/ia.25.2.2

A. TOLSTYKH, A. GOLUBINSKIY

DATA MINING BASED ON DEEP REINFORCEMENT LEARNING FOR PREDICTION OF OPERATING FREQUENCIES AND BANDS IN A COGNITIVE RADIO SYSTEM

Tolstykh A., Golubinskiy A. **Data Mining Based on Deep Reinforcement Learning for Prediction of Operating Frequencies and Bands in a Cognitive Radio System.**

Abstract. The paper proposes a method for solving the problem of choosing a communication channel in cognitive radio based on information about the current state of all available communication channels using the mathematical apparatus of reinforcement learning. The method consists in formalizing the problem of choosing communication channels in terms of "environment-agent" and training agents using the REINFORCE, SARSA and A2C algorithms. The calculation of memory costs for solving the problem of selecting communication channels using classical methods is given. The memory estimate is $\mathbf{4\times2^{2n}}$ bytes for a random state of channels (busy/free) and $\mathbf{4\times n^2}$ bytes for one free channel at each step when solving the problem using the tabular Q-learning algorithm. Two different formalizations of the reward for the agent within the framework of the problem being solved using reinforcement learning are presented – for the trivial case (binary availability/unavailability of the frequency channel) and for a more complex case considering the power (in dB) in the selected communication channel. The restriction on the first formalization is that at each iteration there should be only one free communication channel out of all available channels. The second proposed formalization of the reward function does not impose such restrictions and is more universal. Computational experiments are presented for the corresponding formalizations of the reward function. Agents are trained using the SARSA and A2C algorithms. On average, error-free solutions are achieved after 8,000 training episodes for the corresponding formalizations of training in a model problem for various agent implementations. The REINFORCE algorithm does not provide error-free solutions, but reward formulation takes into account the improved training efficiency of the REINFORCE algorithm. Theoretical estimates of the computational complexity of the considered methods are provided, which are consistent with the computational experiments.

Keywords: cognitive radio, reinforcement learning, deep learning, artificial neural network, multilayer perceptron, reward function, software-defined radio, synthetic data, augmentation, artificial intelligence.

References

1. Tong W., Zhu P. Seti 6G. Put' ot 5G k 6G glazami razrabotchikov. Ot podklyuchennykh lyudey i veshchey k podklyuchennomu intellektu [6G networks. The path from 5G to 6G through the eyes of developers. From connected people and things to connected intelligence]. Moscow: DMK Press, 2022. 624 p. (In Russ.).
2. Fette B.-A. Cognitive Radio Technology. Elsevier. 2006. 622 p.
3. Komashinskiy V., Smirnov D. Neyronnye seti i ikh primenenie v sistemakh upravleniya i svyazi [Neural networks and their application in control and communication systems]. Moscow: Hotline–Telecom, 2003. 93 p. (In Russ.).
4. Golubinskiy A., Tolstykh A. [Hybrid method for training convolutional neural networks]. Informatika i avtomatizatsiya – Informatics and Automation. 2021. vol. 20. no. 2. pp. 463–490. DOI: 10.15622/ia.2021.20.2.8. (In Russ.).
5. Wu C., Chowdhury K.-R., Di Felice M., Meleis M. Spectrum Management of Cognitive Radio Using Multi-Agent Reinforcement Learning. 9th International Conference

on Autonomous Agents and Multiagent Systems. 2010. vol. 1–3. pp. 1705–1712. DOI: 10.1145/1838194.1838199.
6. Kiran U., Kumar P.-D., Reddy R.-K., Ranjith M. Efficient Exploration for Reinforcement Learning Based Distributed Spectrum Sharing in Cognitive Radio System. International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering. 2013. vol. 2. no. 11. pp. 5596–5604.
7. Yau K.-L.-A., Poh G.-S., Chien S.-F., Al-Rawi1 H.-A.-A. Application of Reinforcement Learning in Cognitive Radio Networks: Models and Algorithms. The Scientific World Journal. 2014. vol. 1. pp. 1–23. DOI: 10.1155/2014/209810.
8. Abolarinwa J.-A., Latiff A.-N.-M. Channel Decision in Cognitive Radio Enabled Sensor Networks A Reinforcement Learning Approach. International Journal of Engineering and Technology (IJET). 2015. vol. 7. no. 4. pp. 1394–1404.
9. Raj V., Dias I., Tholeti T., Kalyani S. Spectrum Access In Cognitive Radio Using A Two Stage Reinforcement Learning Approach. IEEE. 2018. vol. 12. no. 1. pp. 20–34. DOI: 10.1109/JSTSP.2018.2798920.
10. Tubachi S., Venkatesan M., Kulkarni A.-V., et al. Predictive learning model for Cognitive Radio using Reinforcement Learning. IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI). 2017. pp. 564–567. DOI: 10.1109/ICPCSI.2017.8391775.
11. Jang S.-J., Han C.-H., Lee K.-E., et al. Reinforcement learning-based dynamic band and channel selection in cognitive radio ad-hoc networks. J Wireless Com Network. 2019. vol. 2019. pp. 1–25. DOI: 10.1186/s13638-019-1433-1.
12. Singhal C., Thanikaiselvan V. Cross Layering Using Reinforcement Learning In Cognitive Radio-Based Industrial Internet Of Ad-Hoc. International Journal of Computer Networks & Communications (IJCNC). 2022. vol. 14. no. 4. pp. 1–17. DOI: 10.5121/ijcnc.2022.14401.
13. Talekar S., Banait S., Patil M. Improved Q-Reinforcement Learning Based Optimal Channel Selection In Cognitive Radio Networks. International Journal of Computer Networks & Communications (IJCNC). 2023. vol. 15. no. 3. pp. 1–14. DOI: 10.5121/ijcnc.2023.15301.
14. Rosen D., Rochez I., McIrvin C., Lee J., D'Alessandro K., Wiecek M., et al. RFRL Gym_A Reinforcement Learning Testbed for Cognitive Radio Applications. International Conference on Machine Learning and Applications (ICMLA). 2023. pp. 279–286. DOI: 10.1109/ICMLA58977.2023.00046.
15. Report (International Telecommunication Union Telecommunication Sectors) ITU-R SM.2152 (09/2009) [Definitions of Software Defined Radio (SDR) and Cognitive Radio System (CRS)]. (In Russ.).
16. Chitnavis S., Kwasinski A. Cross Layer Routing in Cognitive Radio Network Using Deep Reinforcement Learning. IEEE Wireless Communications and Networking Conference (WCNC). 2019. pp. 1–13. DOI: 10.1109/WCNC.2019.8885918.
17. Obite F., Usman A.-D., Okafor E. An overview of deep reinforcement learning for spectrum sensing in cognitive radio networks. Digital Signal Processing. 2021. vol. 113. pp. 1–18. DOI: 10.1016/j.dsp.2021.103014.
18. Tondwalkar A., Kwasinski A. Deep Reinforcement Learning for Distributed and Uncoordinated Cognitive Radios Resource Allocation. 2022. pp. 1–13. arXiv: 2205.13944v1.
19. Sutton R., Barto E. Obuchenie s podkrepleniem [Reinforcement learning]. Moscow: DMK Press, 2020. 552 p. (In Russ.).
20. Gresser L., Keng V. Glubokoye obuchenie s podkrepleniem: teoria i praktika na yazyke Python [Deep Reinforcement Learning: Theory and Practice in Python]. SPB: Peter. 2022. 416 p.

21. Wang H., Yu Y. Exploring Multi-Action Relationship in Reinforcement Learning. Springer, Cham. 2016. pp. 1–13. DOI: 10.1007/978-3-319-42911-3_48.
22. Kingma D., Ba J. Adam: A Method for Stochastic Optimization. CoRR. 2014. T. abs/1412.6980.

Tolstykh Andrey — Ph.D., Software engineer, OOO “RTK”. Research interests: artificial neural networks, machine learning, reinforcement learning. The number of publications — 65. tolstykh.aa@yandex.ru; 1, Vysokovoltny Ave., 127566, Moscow, Russia.

Golubinskiy Andrey — Ph.D., Dr.Sci., Associate Professor, Head of Department, Russian Science Foundation (RSF). Research interests: machine learning, neural network modeling, automated control systems with artificial intelligence elements, speech signal processing. The number of publications — 250. annikgol@mail.ru; 14, Solyanka St., 109240, Moscow, Russia; office phone: +7 (910) 346-6537.