О.П. МОСАЛОВ, Д.В. ПРОХОРОВ, В.Г. РЕДЬКО
Институт оптико-нейронных технологий РАН,
Московский физико-технический институт,
Toyota Technical Center, Ann Arbor, MI, USA.
olegmos_@mail.ru, dprokhorov@ttc-usa.com, redko@iont.ru

СРАВНЕНИЕ ЭВОЛЮЦИИ И ОБУЧЕНИЯ КАК МЕТОДОВ АДАПТАЦИИ АГЕНТОВ*

Аннотация

Рассматривается модель популяции агентов-брокеров, адаптация которых может производиться как с помощью обучения, так и эволюции. Проводится сравнение эффективности этих двух методов адаптации. Показано, что несмотря на то, что обучение самостоятельно не может найти оптимальную стратегию поведения, оно помогает эволюции найти эту стратегию быстрее.

Изучение автономных адаптивных агентов – важное направление исследований в вычислительном интеллекте (Computational Intelligence). Такие агенты, подобно живым организмам, могут обладать собственными целями, собственными знаниями, формировать собственную политику поведения, выполнять те или иные действия, а также взаимодействовать друг с другом. В настоящей работе исследуется модель автономных агентов, которые могут адаптироваться как с помощью обучения, так и эволюции.

1. Агент и его система управления

Рассматриваем агента-брокера, который имеет ресурсы двух типов: деньги и акции, сумма этих ресурсов составляет капитал агента $C(t)$, доля акций в капитале равна $u(t)$. Внешняя среда определяется временным рядом $X(t)$, $t = 0,1,2,...$, $X(t)$ – курс акций на бирже в момент времени t. Агент стремится увеличить свой капитал $C(t)$, изменяя значение $u(t)$. Динамика капитала определяется выражением [1]:

$$C(t+1) = C(t)\,\{1 + u(t+1)\,\Delta X(t+1) / X(t)\}\,[1 - J\,|u(t+1) - u(t)|], \qquad (1)$$

* Работа выполнена частичной поддержке программы Президиума РАН «Интеллектуальные компьютерные системы» (проект 2-45) и РФФИ (проект № 04-01-00179).

где $\Delta X(t+1) = X(t+1) - X(t)$ – текущее изменение курса акций, J – параметр, учитывающий расходы агента на покупку/продажу акций. Следуя [2], используем логарифмическую шкалу для ресурса агента, $R(t) = \log C(t)$. Текущее подкрепление агента $r(t) = R(t+1) - R(t)$ равно:

$$r(t) = \log \{1 + u(t+1)\, \Delta X(t+1) / X(t)\} + \log [1 - J\, |u(t+1) - u(t)|]. \qquad (2)$$

Полагаем, что переменная u может принимать только два значения $u = 0$ (весь капитал в деньгах) или $u = 1$ (весь капитал в акциях).

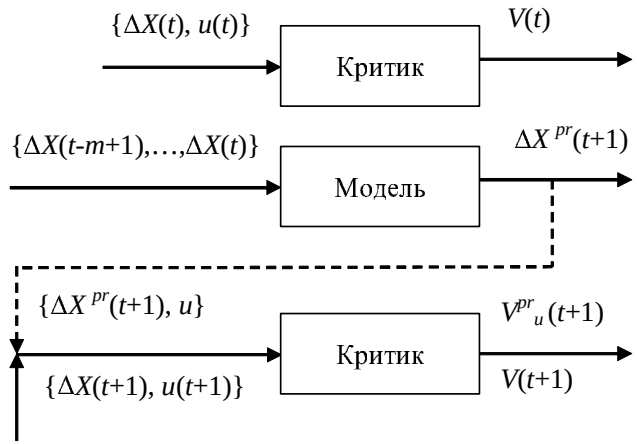

Рис. 1. Схема системы управления агента. НС Критика показана для двух последовательных тактов времени.

2. Алгоритм обучения

Система управления агента представляет собой простой адаптивный критик, состоящий из двух нейронных сетей (НС): Модель и Критик (рис. 1). Цель адаптивного критика – максимизировать функцию полезности $U(t)$, определяемую выражением

$$U(t) = \sum_{j=0}^{\infty} \gamma^j r(t + j), \qquad t = 1,2,...,$$

Делая разумное предположение $\Delta X(t) << X(t)$, полагаем, что ситуация $\mathbf{S}(t)$, характеризующая состояние агента, зависит только от двух величин, $\Delta X(t)$ и $u(t)$: $\mathbf{S}(t) = \{\Delta X(t), u(t)\}$.

Модель предназначена для прогнозирования изменения курса временного ряда. На вход Модели подается m предыдущих значений изменения курса $\Delta X(t\text{-}m\text{+}1), \ldots, \Delta X(t)$, на выходе формируется прогноз изменения курса в следующий такт времени $\Delta X^{pr}(t\text{+}1)$. Модель представляет собой двухслойную НС, работа которой описывается формулами:

$$\mathbf{x^M} = \{\Delta X(t\text{-}m\text{+}1), \ldots, \Delta X(t)\}, \quad y^M{}_j = \text{th}\,(\textstyle\sum_i w^M{}_{ij}\, x^M{}_i), \qquad \Delta X^{pr}(t\text{+}1) = \textstyle\sum_j v^M{}_j\, y^M{}_j\,,$$

где $\mathbf{x^M}$ – входной вектор, $\mathbf{y^M}$ – вектор выходов нейронов скрытого слоя, $w^M{}_{ij}$ и $v^M{}_j$ – веса синапсов НС.

Критик предназначен для оценки качества ситуаций $V(\mathbf{S})$, а именно, оценки функции полезности $U(t)$ для агента, находящегося в рассматриваемой ситуации $\mathbf{S}$. Критик представляет собой двухслойную НС, работа которой описывается формулами:

$$\mathbf{x^C} = \mathbf{S}(t) = \{\Delta X(t), u(t)\}, \quad y^C{}_j = \text{th}\,(\textstyle\sum_i w^C{}_{ij}\, x^C{}_i),\; V(t) = V(\mathbf{S}(t)) = \textstyle\sum_j v^C{}_j\, y^C{}_j\,,$$

где $\mathbf{x^C}$ – входной вектор, $\mathbf{y^C}$ – вектор выходов нейронов скрытого слоя, $w^C{}_{ij}$ и $v^C{}_j$ – веса синапсов НС.

Каждый момент времени t выполняются следующие операции:

1) Модель предсказывает следующее изменение временного ряда $\Delta X^{pr}(t\text{+}1)$.

2) Критик оценивает величину V для текущей ситуации $V(t) = V(\mathbf{S}(t))$ и для предсказываемых ситуаций для обоих возможных действий $V^{pr}{}_u(t\text{+}1) = V(\mathbf{S^{pr}}_u(t\text{+}1))$, где $\mathbf{S^{pr}}_u(t\text{+}1) = \{\Delta X^{pr}\,(t\text{+}1), u\}$, $u = 0$ либо $u = 1$.

3) Применяется ε-жадное правило: действие, соответствующее максимальному значению $V^{pr}{}_u(t\text{+}1)$ выбирается с вероятностью $1 - \varepsilon$, и альтернативное действие выбирается с вероятностью ε ($0 < \varepsilon << 1$). Выбор действия есть выбор величины $u(t\text{+}1)$: перевести весь капитал в деньги: $u(t\text{+}1) = 0$; либо в акции: $u(t\text{+}1) = 1$.

4) Выбранное действие $u(t\text{+}1)$ выполняется. Происходит переход к моменту времени $t\text{+}1$. Подсчитывается подкрепление $r(t)$ согласно (2). Наблюдаемое значение $\Delta X(t\text{+}1)$ сравнивается с предсказанием $\Delta X^{pr}(t\text{+}1)$. Веса НС Модели подстраиваются так, чтобы минимизировать ошибку предсказания методом обратного распространения ошибки. Скорость обучения Модели равна $\alpha_M > 0$.

5) Критик подсчитывает $V(t\text{+}1) = V(\mathbf{S}(t\text{+}1))$, $\mathbf{S}(t\text{+}1) = \{\Delta X(t\text{+}1), u(t\text{+}1)\}$. Рассчитывается ошибка временной разности [3]:

$\delta(t) = r(t) + \gamma V(t+1) - V(t)$.

Величина $\delta(t)$ характеризует ошибку в оценке $V(t) = V(\mathbf{S}(t))$ – суммарной награды, которую можно получить, исходя из состояния $\mathbf{S}(t)$.

6) Веса НС Критика подстраиваются так, чтобы минимизировать величину $\delta(t)$, это обучение осуществляется градиентным методом, аналогично методу обратного распространения ошибки. Скорость обучения Критика равна $\alpha_C > 0$.

3. Схема эволюции

Эволюционирующая популяция состоит из n агентов. Каждый агент имеет ресурс $R(t)$, который изменяется в соответствии с подкреплениями агента: $R(t+1) = R(t) + r(t)$.

Эволюция происходит в течение ряда поколений, n_g=1,2,…, N_g. Продолжительность каждого поколения n_g равна T тактов времени (T – длительность жизни агента). В начале каждого поколения начальный ресурс каждого агента равен нулю, т.е., $R(T(n_g\text{-}1)+1) = 0$.

Начальные веса синапсов обоих НС (Модели и Критика) формируют геном агента $\mathbf{G}=\{\mathbf{W}_{M0},\mathbf{W}_{C0}\}$. Геном $\mathbf{G}$ задается в момент рождения агента и не меняется в течение его жизни.

В конце каждого поколения определяется агент, имеющий максимальный ресурс R_{max} (n_g) (лучший агент поколения n_g). Этот лучший агент порождает n потомков, которые составляют новое (n_g+1)-ое поколение. Геномы потомков $\mathbf{G}$ отличаются от генома родителя небольшими мутациями.

Более конкретно, в начале каждого нового (n_g+1)-го поколения мы полагаем для каждого агента $G_i(n_g+1) = G_{best,i}$ (n_g) + rand_i, $\mathbf{W}_0(n_g+1) = \mathbf{G}(n_g+1)$, где $\mathbf{G}_{\mathbf{best}}(n_g)$ – геном лучшего агента предыдущего n_g-го поколения и rand_i – это $N(0, P^2_{mut})$, т.е., нормально распределенная случайная величина с нулевым средним и стандартным отклонением P_{mut} (интенсивность мутаций), которая добавляется к каждому весу.

Таким образом, геном $\mathbf{G}$ (начальные веса синапсов, получаемые при рождении) изменяется только посредством эволюции, в то время как текущие веса синапсов $\mathbf{W}$ дополнительно к этому подстраиваются посредством обучения. При этом в момент рождения агента $\mathbf{W} = \mathbf{W}_0 = \mathbf{G}$.

4. Результаты моделирования

Изложенная модель была реализована в виде компьютерной программы. В качестве временного ряда использовались синусоида:

$$X(t) = 0{,}5(1 + \sin(2\pi t/20)) + 1$$

и стохастический временной ряд [1]:

$$X(t) = \exp(p(t)/1200), \quad p(t) = p(t\text{-}1) + \beta(t\text{-}1) + k\,\lambda(t), \quad \beta(t) = \alpha\beta(t\text{-}1) + \mu(t),$$

где $\lambda(t)$ и $\mu(t)$ – два нормальных процесса с нулевым средним и единичной дисперсией, $\alpha = 0{,}9$, $k = 0{,}3$.

Параметры расчета составляли: фактор забывания $\gamma = 0{,}9$; количество входов НС Модели $m = 10$; количество нейронов в скрытых слоях НС Модели и Критика $N_{hM} = N_{hC} = 10$; скорость обучения Модели и Критика $\alpha_M = \alpha_C = 0{,}01$; параметр ε-жадного правила $\varepsilon = 0{,}05$; интенсивность мутаций $P_{mut} = 0{,}1$; расходы агента на покупку/продажу акций $J = 0$, продолжительность поколения $T = 200$, численность популяции $n = 10$.

Мы анализировали следующие варианты рассматриваемой модели:
Случай L (чистое обучение), в этом случае рассматривался отдельный агент, который обучался методом ошибки временной разности;
Случай E (чистая эволюция), т.е. рассматривается эволюционирующая популяция без обучения;
Случай LE (обучение + эволюция), т.е. эволюция популяции агентов, которые обучаются в течение своей жизни.

Было проведено сравнение ресурса, приобретаемого агентами за 200 временных тактов для этих трех способов адаптации. Для случаев E и LE бралось $T = 200$ и регистрировалось максимальное значение ресурса в популяции $R_{max}(n_g)$ в конце каждого поколения. В случае L (чистое обучение) рассматривался только один агент, ресурс которого для удобства сравнения со случаями E и LE обнулялся каждые $T = 200$ тактов времени: $R(T(n_g\text{-}1)+1) = 0$. В этом случае индекс n_g увеличивался на единицу после каждых T временных тактов, и полагалось $R_{max}(n_g) = R(Tn_g)$.

Графики $R_{max}(n_g)$ для синусоиды показаны на рис. 2. Чтобы исключить уменьшение значения $R_{max}(n_g)$ из-за случайного выбора действий при применении ε-жадного правила для случаев LE и L, полагалось $\varepsilon = 0$ после $n_g = 100$ для случая LE и после $n_g = 2000$ для случая L (на рис. 2 видно резкое увеличение $R_{max}(n_g)$ после $n_g = 100$ и $n_g = 2000$ для соответствующих случаев). Результаты усреднены по 1000 экспериментам.

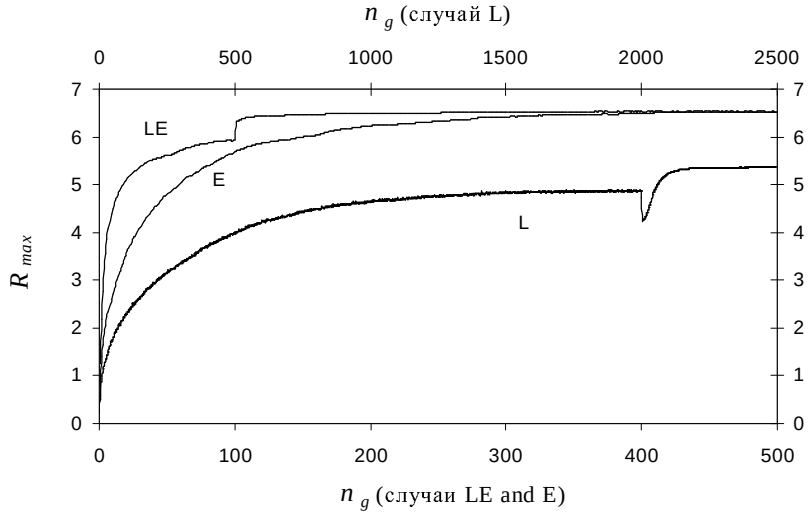

Рис. 2. Зависимости $R_{max}(n_g)$. Кривая LE соответствует случаю эволюции, объединенной с обучением, кривая E – случаю чистой эволюции, кривая L – случаю чистого обучения. Временная шкала для случаев LE и E (номер поколения n_g) представлена снизу, для случая L (индекс n_g) – сверху. Моделирование проведено для синусоиды, кривые усреднены по 1000 экспериментам.

Анализ экспериментов, представленных на рис. 2, показывает, что в случаях LE (обучение + эволюция), и E (чистая эволюция) находится оптимальная стратегия – переводить капитал в акции/деньги при росте/падении курса $X(t)$. Это соответствует асимптотическому значению ресурса $R_{max}(500) = 6{,}5$.

В случае L (чистое обучение) асимптотическое значение ресурса ($R_{max}(2500) = 5{,}4$) существенно меньше. Анализ экспериментов для этого случая показывает, что одно обучение обеспечивает нахождение только следующей «субоптимальной» стратегии поведения: агент держит капитал в акциях при росте и при слабом падении курса и переводит капитал в деньги при сильном падении курса.

Итак, результаты, представленные на рис. 2, демонстрируют, что хотя обучение в настоящей модели и несовершенно, оно способствует более быстрому нахождению оптимальной стратегии поведения по сравнению со случаем чистой эволюции.

Система управления агента включает в себя Модель, предназначенную для предсказания изменения ΔX (t+1) временного ряда в следующий такт времени t+1. Мы проанализировали работу Модели и обнаружили очень интересную особенность. Модель может давать неверные предсказания (рис. 3), однако агент, тем не менее, может использовать эти предсказания для принятия верных решений – стратегия поведения агентов при этих прогнозах практически оптимальна. Отметим, что для зависимостей, представленных на рис. 3, форма предсказанной кривой $\Delta X^{pr}(t+1)$ близка по форме к реальной зависимости ΔX (t+1), но отличается от нее знаком и величиной.

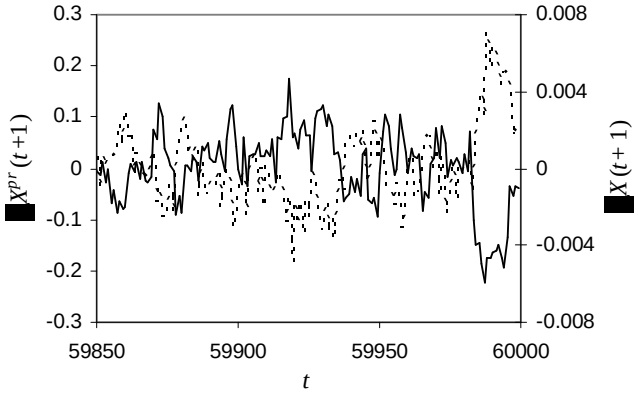

Рис. 3. Предсказываемые $\Delta X^{pr}(t+1)$ (пунктирная линия) и реальные изменения $\Delta X(t+1)$ (сплошная линия) стохастического временного ряда. Случай эволюции, объединенной с обучением. Кривые ΔX^{pr} (t+1) и ΔX (t+1) различаются как по величине, так и знаком.

По-видимому, наблюдаемое увеличение значений ΔX^{pr} нейронной сетью Модели «полезно» для работы нейронной сети Критика, так как реальные значения $\Delta X(t+1)$ слишком малы (порядка 0,001). Таким образом, нейронная сеть Модели может не только предсказывать значения $\Delta X^{pr}(t+1)$, но также осуществлять полезные преобразования этих значений. Изменение знака $\Delta X^{pr}(t+1)$ по сравнению с $\Delta X(t+1)$, по-видимому, не принципиально для работы Критика.

Эти особенности работы нейронной сети Модели обусловлены доминирующей ролью эволюции над обучением при оптимизации системы управления агентов. Это делает предпочтительными такие системы

управления, которые устойчивы в эволюционном смысле. Кроме того, важно подчеркнуть, что задача, которую «решает» эволюция в настоящей модели, значительно проще, чем та задача, которую решает обучение. Эволюции достаточно обеспечить выбор действий (покупать или продавать), приводящий к награде. А схема обучения предусматривает довольно сложную процедуру прогноза ситуации **S,** оценки качества прогнозируемых ситуаций, итеративного формирования оценок качества ситуаций $V(\mathbf{S})$ и выбора действия на основе этих оценок. То есть эволюция идет к нужному результату более прямым путем, а так как задача агентов проста, то эволюция в определенной степени «задавливает» довольно сложный механизм обучения.

5. Выводы

Построена модель популяции автономных агентов, система управления которых основана на нейросетевых адаптивных критиках. Показано, что хотя обучение в настоящей модели и несовершенно, оно способствует более быстрому нахождению оптимальной стратегии поведения для случая комбинации обучения и эволюции по сравнению со случаем чистой эволюции.

Кроме того, проведенное моделирование демонстрирует, что сложные нейросетевые схемы обучения могут быть эволюционно нестабильны, если процесс обучения неустойчив относительно к возмущениям весов синапсов нейронных сетей.

Список литературы

1. Prokhorov, D., Puskorius, G., Feldkamp, L.: Dynamical Neural Networks for Control. In J. Kolen and S. Kremer (eds.) A Field Guide to Dynamical Recurrent Networks. IEEE Press, (2001) 23-78.
2. Moody, J., Wu, L., Liao, Y., Saffel, M.: Performance Function and Reinforcement Learning for Trading Systems and Portfolios. Journal of Forecasting, 17 (1998) 441-470.
3. Sutton, R., Barto, A.: Reinforcement Learning: An Introduction. Cambridge: MIT Press (1998).