

МНОГОАГЕНТНЫЙ НЕЙРОЭВОЛЮЦИОННЫЙ ПОДХОД К АДАПТИВНОМУ УПРАВЛЕНИЮ

Цой Ю.Р.

Томский политехнический университет, Кафедра
вычислительной техники,

E-mail: neuroevolution@mail.ru

Аннотация

В работе описывается один из возможных вариантов построения многоагентной системы (МАС), использующей нейроэволюционный (НЭ) подход. Рассматривается применение этой системы для задачи преследования жертвы (prey capture).

Введение

Эволюция демонстрирует выдающийся пример адаптации к постоянно меняющимся условиям окружающей среды. Благодаря наследственности и изменчивости работают механизмы приспособления видов в природе. Компьютерные алгоритмы, использующие методы, основанные на таких природных принципах, получили название эволюционных. Одной из их самых распространенных разновидностей являются генетические алгоритмы (ГА). Суть их работы заключается в том, что в течение некоторого числа поколений развивается популяция потенциальных решений. Если в самом начале все особи сгенерированы, как правило, случайным образом, и соответствующие им решения являются в большинстве случаев неудовлетворительными, то по прошествии определенного времени часто можно наблюдать улучшение получаемых решений, по крайней мере, среди лучших представителей популяции.

Управление также требует проявления гибкости и некоторых способностей к адаптации. Одним из подходов, обеспечивающих приспособительные возможности, является многоагентная архитектура. В системах управления, как правило, используется правило (алгоритм) или набор правил, ответственный за регулировку и подстройку параметров. Встречающиеся решения с несколькими алгоритмами, выбор между которыми осуществляется в зависимости от ситуации, как правило, не способны значительно модифицировать сами алгоритмы. В МАС используемые правила динамически изменяются, таким образом, становится возможным отслеживание изменений окружающей среды. Агент представляет некоторое правило управления системой и в каждый момент времени конкурирует с другими агентами-правилами, при этом правило, определяемое агентом, может измениться на следующем шаге [3].

Одно из основных применений искусственных нейронных сетей (ИНС) – адаптивное управление. Однако, для использования ИНС необходимо определить её топологию, что подчас является непростой задачей. Один из возможных выходов –

использование ГА для настройки структуры нейросети, получил название нейроэволюция.

Общее описание архитектуры предлагаемой МАС

Имеется популяция из N агентов $A = \{A_i \mid i \in [1; N]\}$. Каждый агент представляет ИНС, заданную перечислением всех её связей. Нейроны сети индексируются в зависимости от занимаемого ими места. Входные нейроны получают индексы от 0 до $(N_I - 1)$, где N_I – число входов сети, выходные – от N_I до $(N_I + N_O - 1)$, где N_O – число выходов, скрытые нейроны – от $(N_I + N_O)$ до $(N_N - 1)$, где N_N – суммарное число нейронов в сети. Связь описывается следующим образом: $c = \{(b, e, w) \mid b \in [0; N_I - 1] \cup [N_I + N_O; N_N - 1], e \in [N_I; N_N - 1], w \in [-65,536; 65,535]\}$, здесь b и e – соответственно индексы начального и конечного нейронов связи, w – вес связи. Таким образом, каждый агент может быть определен как $A_i = \{c_{ij} \mid j \in [1; N_{C,i}]\}$, где $N_{C,i}$ – число связей в ИНС i -го агента.

Также есть система S , обладающая набором параметров. Общая цель МАС – управление системой S таким образом, чтобы она достаточно эффективно выполняла свои функции. Действие i -го агента на систему описывается в соответствии с формулой $S_k' = C(A_i, S_k)$, где S_k и S_k' – соответственно состояния системы S до управления и после него, C – функция, реализующая управление системой под руководством агента A_i . Сама система S находится в некоторой изменчивой среде E , также имеющей ряд характеристик, которые могут изменяться как самопроизвольно, так и под влиянием изменений в S . Опишем изменение среды $E_k' = V(E_k, S)$, где E_k' и E_k – соответственно состояния среды до и после изменения, V – функция, определяющая закон взаимодействия системы S со средой, а также правила изменчивости самой среды.

При оценивании популяции каждый агент имеет возможность управлять системой, причем после завершения его работы система возвращается в состояние, предшествовавшее управлению. В результате каждому агенту ставится в соответствие число, характеризующее его успешность (приспособленность), т.е. существует функция $f(C(A_i, S(t)), E(t)) = f_i$, оценивающая работу i -го агента, где $S(t)$ и $E(t)$ – соответственно состояния системы и среды в текущем поколении. Затем агенты сортируются в порядке ухудшения этой характеристики. Самый лучший агент изменяет параметры управляемой системы и переходит в новое поколение без изменений. Все остальные агенты замещаются потомками, получившимися в результате скрещивания «родителей» из текущего поколения. При этом к скрещиванию допускается только те агенты, чья характеристика лучше средней, т.е. если A' – агенты отобранные для скрещивания и f_{cp} – средняя величина приспособленности, то $A' = \{A_i \mid f_i > f_{cp}\}$. При скрещивании происходит рекомбинация топологий ИНС, представленных

агентами, в результате чего есть возможность получить новые структурные решения. Это также достигается с помощью механизма мутации потомков, «точно» изменяющего параметры ИНС.

Таким образом, $A(t+1) = A_0(t) \cup R(A'(t))$, где t – номер поколения, $A(t+1)$ – популяция на $(t+1)$ поколении, $A_0(t)$ – лучшая особь в поколении t , R – функция, генерирующая подпопуляцию потомков, полученных после применения к $A'(t)$ операторов кроссовера (скрещивание) и мутации. Изменение системы: $S(t+1) = C(A_0(t), S(t))$, где $S(t)$ и $S(t+1)$ – соответственно состояния S в поколениях t и $(t+1)$. Аналогично для среды $E(t+1) = V(E(t), S(t+1))$, т.е. среда реагирует на действия системы S , управляемой $A_0(t)$. Стоит отметить, что в явном виде обратная связь в поведении отсутствует, поскольку выбор следующего действия системы S не зависит от предыдущего, т.е. можно говорить о том, что рассматривается система без памяти.

Постановка задачи

Задача преследования жертвы является одной из классических для исследования адаптивного поведения [1]. При этом управляемым считается хищник, а жертва действует на основании алгоритма, который является недискретной модификацией алгоритма из [2]. Жертва выбирает направление движения, чтобы как можно дальше удалиться от преследователя. Скорость жертвы постоянна и равна 0,475. Управление движением хищника осуществлялось с использованием информации о векторе «хищник→жертва». Управляющими параметрами являлись угол, на который должен повернуться хищник и расстояние, которое он должен пройти в заданном направлении в диапазоне $[0; 1]$. Каждый управляющий агент представляет ИНС с 2 входами и 2 выходами, при этом топология сети у каждого агента может быть различной.

Ставилось два различных вида экспериментов. В первом жертва неподвижна, а во втором жертва двигалась в соответствии с разработанным алгоритмом. В обоих случаях хищник должен подойти на расстояние не больше 0,5 единиц. В самом начале эксперимента хищник находится в начале координат, а жертва в случайной точке в пределах определенного радиуса r (были использованы значения 10 и 25). В каждом запуске алгоритма производилось 250 испытаний МАС с различными стартовыми позициями, причем после успешного завершения одного испытания популяция агентов не изменяясь подвергалась следующему испытанию. В качестве критерия оценки запуска для случая неподвижной жертвы было взято отношение

$$\eta_1 = \frac{Length_{SUM}}{Steps_{SUM}} \quad (1)$$

где $Length_{SUM}$ – суммарное расстояние, которое робот прошел за 250 испытаний, $Steps_{SUM}$ – потраченное для этого количество шагов. Т.е. оценивается отношение фактического расстояния, прой-

денного хищником к максимальному теоретическому. Для случая подвижной жертвы добавилась ещё одна оценка

$$\eta_2 = \frac{\Delta Length_{SUM}}{Length_{0,SUM}} \quad (2)$$

где $Length_{0,SUM}$ – суммарное начальное расстояние от хищника до жертвы, $\Delta Length_{SUM}$ – суммарное расстояние, которое жертва успела пройти от своей начальной позиции. Вторым критерием оценивается, насколько далеко ушла жертва в зависимости от своего первоначального удаления от хищника. Для каждого случая осуществлялось 10 запусков, результаты усреднялись.

Результаты экспериментов

Значения критерия η_1 для неподвижной жертвы при различных значениях размера популяции ($|A|$) и радиуса удаления (r) приведены в табл.1. В последних двух столбцах указаны значения при предварительном обучении популяции перед каждым испытанием в течение 5 поколений.

Табл.1.

	$r = 10$	$r = 25$	$r = 10$ (pre)	$r = 25$ (pre)
$ A = 25$	0,6902	0,7609	0,7115	0,7792
$ A = 50$	0,8310	0,9031	0,8559	0,9032

Результаты для случая подвижной жертвы представлены в табл.2 ($r = 10$).

Табл.2.

	η_1	η_2
$ A = 25$	0,5164751	7,291877
$ A = 50$	0,5741533	3,699684

Из данных в табл.2 видно, что использование в два раза большей популяции агентов способствует в два раза более успешной поимке жертвы.

Число случаев, когда количество скрытых нейронов в полученных решениях превышало 1, составляет 10%.

Заключение

Предложенная задача является по большому счету «синтетической» и не требует сложных алгоритмов для решения, однако она вполне подходит, чтобы показать в действии разработанную МАС. Результаты показали, что в работе МАС присутствуют как достоинства (адаптивность, сравнительно простые топологии ИНС), так и недостатки (довольно большая кривизна траектории движения хищника, отсутствие «памяти»).

Источники

1. F. Gomez and R.Miikkulainen. Incremental evolution of complex general behavior. Adaptive behavior, 5:317–342, 1997.
2. Lin, L.-J. Self-improving reactive agents based on reinforcement learning, planning and teaching. Machine Learning, 8:293–321, 1992.
3. Клышинский Э.С. ОДНА МОДЕЛЬ ПОСТРОЕНИЯ АГЕНТА. 1999.