

## **ОБ ЭФФЕКТИВНОМ РАСПРЕДЕЛЕНИИ РЕСУРСОВ НЕЙРЭВОЛЮЦИОННОГО АЛГОРИТМА**

**Дягель А.В.**

**научный руководитель доктор.техн. наук Семенкин Е.С.  
*Сибирский федеральный университет, Институт математики и  
фундаментальной информатики***

Выбор топологии и настройка весов связей искусственной нейронной сети (ИНС) являются главными этапами при использовании нейросетевых технологий для решения практических задач. От этих этапов напрямую зависит эффективность полученной нейросетевой модели и построенных с ее использованием интеллектуальных информационных систем (идентификации, классификации, управления и т.п.). В рамках так называемого нейроэволюционного подхода оба этих этапа осуществляются с помощью эволюционных алгоритмов оптимизации, наиболее известных из которых являются генетические алгоритмы (ГА).

ИНС и ГА являются сравнительно молодыми направлениями исследований в области искусственного интеллекта и принятия решений. Обе концепции используют аналогии природных принципов. Использование ГА для одновременной настройки весов и выбора эффективной структуры ИНС называется нейроэволюцией (НЭ). Вопрос о том, нужно ли «тонко» настраивать топологию ИНС в ходе работы нейроэволюционного алгоритма довольно долго остается открытым, поскольку существующие решения и так удовлетворительно справляются со своими задачами. Есть мнение, что можно найти удачную топологию ИНС, которая в дальнейшем быстрее обучается, однако процесс обучения сетей проводится уже на этапе поиска архитектуры, поэтому аргумент отпадает сам собой. Также приводится довод, что найденная топология сети позволит разработчикам в дальнейшем учитывать этот опыт и более эффективно использовать ИНС.

Как уже отмечалось выше, НЭ методы способны одновременно настраивать структуру и веса сети. Это позволяет получать готовые нейросетевые решения, имея только данные из обучающей выборки, и в большинстве случаев не требует от пользователя наличия глубоких знаний из теории ИНС. Тем не менее, в случае обучения с учителем, когда есть наборы входных и соответствующих им выходных данных, а для решения достаточно использовать многослойную сеть прямого распространения, существующие алгоритмы, например, алгоритм сопряженных градиентов, позволяет обучать ИНС за приемлемое время. Использование здесь НЭ подхода мало оправдывает себя, в особенности учитывая значительно большие по сравнению с градиентными методами требования к объему используемой оперативной памяти, т.к. в большинстве случаев используется популяция нейросетей.

Существует много задач, при решении которых с использованием нейросетевых технологий, встает вопрос о выборе структуры ИНС. Зачастую ограничиваются вариантом с регулярной топологией, который не является оптимальным с точки зрения используемых ресурсов компьютера. Кроме этого настройка топологии требует от пользователя достаточно глубокой компетенции в области нейронных сетей. Автоматический выбор структуры сети позволил бы существенно упростить эту проблему. Именно поэтому и используется нейроэволюционный подход.

В рамках НЭ необходимо установить эффективное распределение ресурсиспользуемой вычислительной системы. Ставится вопрос о том, какие конфигурации ГА работают эффективнее. В условиях ограничений на время работы и

на производительность вычислительной системы можно предложить два подхода. В рамках одного из них популяция ГА, выбирающего структуру ИНС, берется большой, а для ГА, настраивающего весовые коэффициенты, выделяется относительно немного вычислительных ресурсов (размер популяции мал и число поколений невелико). При втором подходе популяция ГА, выбирающего эффективную структуру, берется небольшой, а основное время отводится на работу ГА, настраивающего весовые коэффициенты, популяция которого и количество поколений выбираются большими. Т.к. время проработки самого алгоритма пренебрежимо мало, основным показателем вычислительных затрат является количество вычислений функции ошибки ИНС.

В данной работе для проведения сравнительного анализа использована программная система [1], в которой для решения задач аппроксимации реализован выбор структуры ИНС и настройка весовых коэффициентов с помощью ГА, при этом обеспечена возможность выбора генетических операторов, таких как селекция (турнирная, турнирная с элитизмом), скрещивание (одноточечное, двухточечное, равномерное), мутация (низкая, средняя и высокая). Рабочее окно программной системы приведено на рисунке 1.

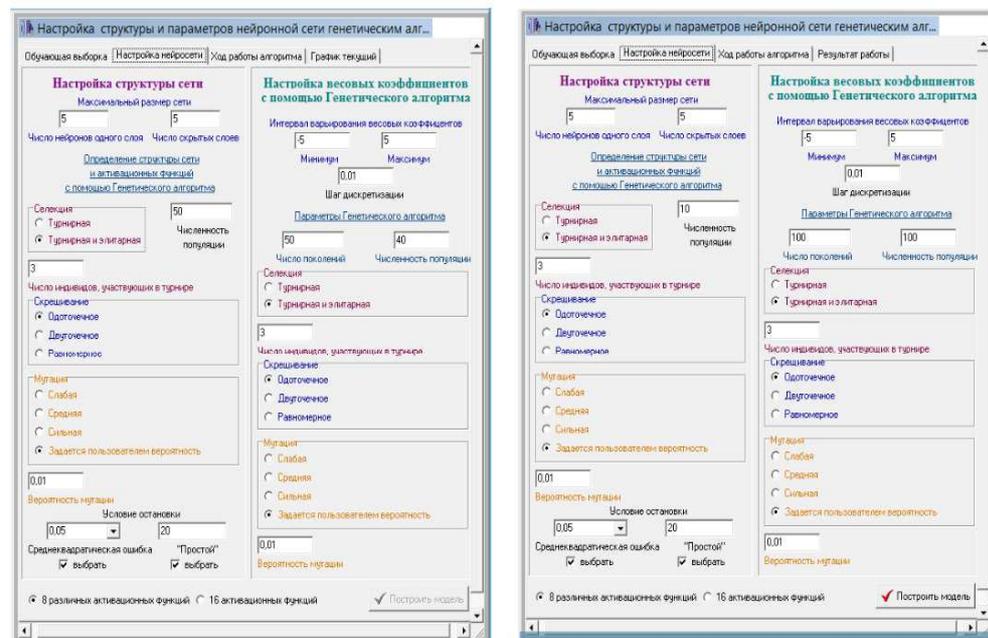


Рисунок 1 - Пример рабочего окна программной системы

В ходе экспериментов использовались конфигурации ГА, установленные разработчиками по умолчанию: селекция турнирная с элитизмом (размер турнира – 3 индивида), скрещивание одноточечное и вероятность мутации 0.01. Вторым подходом реализовывался при (малой) популяции из 10 структур и 100 поколениях по 100 индивидов для ГА, настраивающего весовые коэффициенты (большое количество ресурсов). Для первого подхода были выбраны параметры 50, 50 и 40, соответственно. Для каждого из двух подходов было проведено 5 экспериментов по 20 запусков алгоритма в каждом (т.е. всего по 100 запусков). Условия остановки алгоритма – достижение ошибки на тестовой выборке, равной 0,05, или стагнация (20 поколений

структур без улучшения). Алгоритм прекращал свою работу при выполнении одного из двух критериев. Запуск алгоритма считался удачным, если алгоритм заканчивал свою работу по достижению точности.

Ниже в таблице 1 представлены численные значения оценки затрат для каждого из 5 экспериментов. Данная таблица показывает какой вычислительный ресурс потребовался в ходе экспериментов на вычисление ошибки и количество проверенных структур.

Таблица 1 - Сравнительная таблица затрат на решение задачи

	1 подход		2 подход	
	Число проверенных структур	Число вычислений ошибки	Число проверенных структур	Число вычислений ошибки
Минимальное	100; 50; 100; 10; 50	$2 \cdot 10^5$ ; $1 \cdot 10^5$ ; $2 \cdot 10^5$ ; $3 \cdot 10^5$ ; $1 \cdot 10^5$	20; 20; 10; 20; 20	$2 \cdot 10^5$ ; $2 \cdot 10^5$ ; $1 \cdot 10^5$ ; $2 \cdot 10^5$ ; $2 \cdot 10^5$
Максимальное	1100; 1450; 2450; 1300; 1150	$22 \cdot 10^5$ , 29 * $10^5$ , 49 * $10^5$ , 26 * $10^5$ , 23 * $10^5$	260; 200; 340; 300; 370	$26 \cdot 10^5$ ; 20 * $10^5$ ; 34 * $10^5$ ; 30 * $10^5$ ; 37 * $10^5$
Среднее	517.5; 613.2; 468.4; 626.3; 528.9	990000; 936842,1; 1226315,8; 1 252631,6; 1 057894,7	93,7; 109.5; 111.6; 77.9 ; 107.5	93682.2; 1045000; 1116666.6; 747368.4; 1080000

Из данной таблицы видно, что для достижения требуемой точности второй подход требует в среднем меньших вычислительных затрат (на 25%, примерно). Однако более важной характеристикой нейросетевой модели является точность аппроксимации, поэтому надо оценить ошибку, получаемую лучшей нейросетью, построенной обоими подходами. Такие ошибки приведены в таблице два. В этой таблице пять строк для каждого подхода, соответствующих пяти экспериментам. В каждой ячейке приведена средняя квадратичная ошибка лучшей нейросети, усредненная по 20 прогонам алгоритма.

Таблица 2 – Оценка ошибки аппроксимации

Ошибка аппроксимации		
	1 подход	2 подход
1	0,0016	0,0012
2	0,0011	0,0015
3	0,0013	0,0014
4	0,0015	0,0014
5	0,0025	0,0015

Данные результаты помогут уточнить какой из подходов является более эффективным.

Рабочая гипотеза состояла в том, что при малом количестве структур, хоть и тонко настраивающихся, результаты будут хуже, чем при большом количестве

структур, настраивающихся "грубо". Проверка гипотезы осуществлялась с помощью рангового критерия Вилкоксона[2].

В результате проведенных экспериментов достоверного отличия двух случайных величин не установлено (при пороге значимости 0,05). Таким образом, оба возможных подхода к распределению вычислительных ресурсов нейроэволюционного алгоритма эквивалентны с точки зрения получаемой точности аппроксимации. Т.к. второй из них требует меньшего количества вычислительных ресурсов для достижения требуемой точности (см. табл. 2), то следует предпочитать именно его, т.е. следует предпочитать более тонкую настройку проверяемых структур ИНС, даже если при этом количество самих структур будет невелико.

Отметим, что эффективность нейроэволюционного алгоритма зависит от выбора его конфигурации, чего в нашей работе не было сделано, т.к. это требует многократного решения одной и той же задачи многими генетическими алгоритмами с различными настройками, что не может быть признано удачным подходом для практических задач, обычно требующих серьезных вычислительных ресурсов. Поэтому основным подходом в развитии НЭ моделирования должно быть внедрение в практику самонастраивающихся эволюционных алгоритмов.

Дополнительные наблюдения состоят в том, что совместное использование ЭА и ИНС позволяет решать задачи настройки и обучения ИНС как по отдельности, так и одновременно. Одним из достоинств такого синтезированного подхода является во многом унифицированный подход к решению разнообразных задач классификации, аппроксимации, управления и моделирования. Использование качественной оценки функционирования ИНС позволяет применять НЭ алгоритмы для решения задач исследования адаптивного поведения интеллектуальных агентов, поиска игровых стратегий, обработки сигналов. Несмотря на то, что количество проблем и открытых вопросов, касающихся разработки и применения НЭ алгоритмов (способы кодирования, генетические операторы, методы анализа и др.) велико, часто для успешного решения задачи с использованием НЭ алгоритма достаточно адекватного понимания проблемы и НЭ подхода, свидетельством чего является большое число интересных и успешных работ в данном направлении.

1. Заблоцкий С.Г., Семенкин Е.С. Система эволюционного формирования нейросетевых моделей сложных систем // Компьютерные учебные программы и инновации. - №7. – 2007. – С. 15.

2. Wilcoxon. F.(1945) individual comparisons by ranking methods. Biometrics. L. 80-