

Анализ геометрического метода формирования модульной структуры нейронных сетей

В.Г. Редько

Работа поддержана РФФИ, проект № 04-01-00179

Аннотация

Представлен анализ «геометрического метода» формирования структуры нейронной сети. Метод состоит из специального языка команд-инструкций, на основе которого строятся программы развития структуры и параметров нейронной сети, и эволюционного способа оптимизации программы. Сложная нейронная сеть строится по модульному принципу, и содержит модули как нижних, так и верхних уровней управления, при этом программы формирования модулей разного уровня независимы друг от друга. Метод может быть применен к широкому классу систем автономного адаптивного управления.

1. Общая схема геометрического метода

В конце 1980 годов ряд исследователей начали работы по автоматическому конструированию нейронных сетей на основе эволюционных алгоритмов [1-5]. Однако предложенные в этих работах сравнительно простые методы имели ограниченное применение и не подходили для решения нетривиальных задач, требующих нейронных сетей большого размера.

Для преодоления указанной проблемы сотрудниками Анимат-лаборатории (AnimatLab, Ecole Normale Supérieure, Paris, France), руководимой Жаном-Аркадием Мейером, был предложен геометрический метод формирования модульной структуры нейронных сетей [6-8]. Этот метод позволяет конструировать нейронные сети достаточно сложной структуры. Метод был успешно применен к формированию нейронной сети подвижных 6-ногих роботов.

В настоящей статье представлены описание и анализ данного метода.

Общая схема метода состоит в следующем. Нейронная сеть искусственного организма (анимата) формируется с помощью специальной программы, контролирующей процесс конструирования сети. Эта программа имитирует процесс развития нейронной сети в процессе индивидуального взросления организма. Сама программа оптимизируется с помощью эволюционного алгоритма. Нейронная сеть формируется в двумерной ограниченной области. Программа состоит из инструкций (команд), которые определяют процессы возникновения новых нейронов (или исчезновения уже имеющихся нейронов) в этой области, формирование связей между нейронами и задание весов синаптических связей между нейронами. Инструкции программы составляют геном анимата. Работа формирующихся нейронных сетей оценивается по поведению анимата некоторой естественной функцией приспособленности, которая определяет отбор наиболее эффективных программ, кодируемых геномами аниматов.

Общая схема метода представлена на рис. 1.

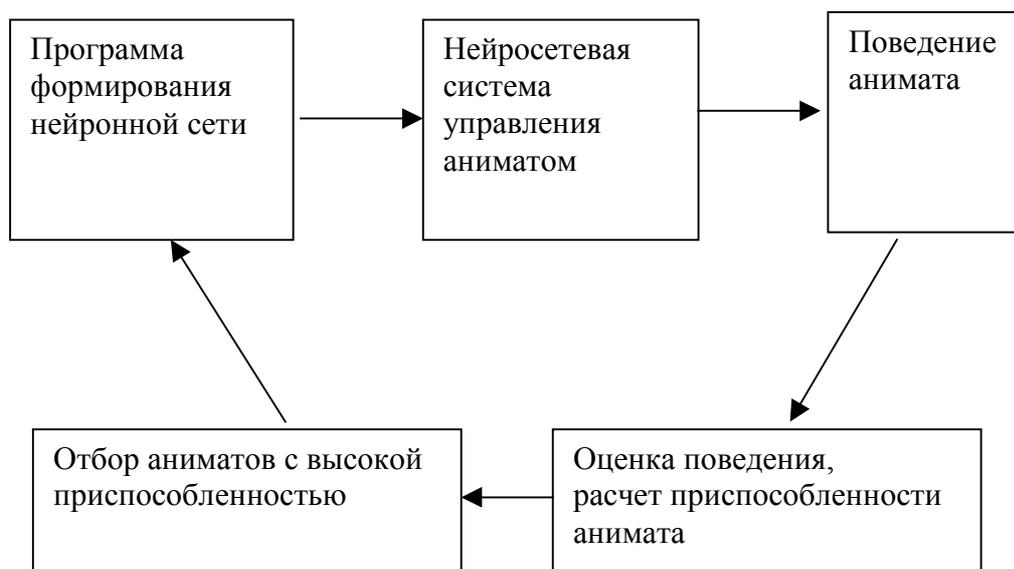


Рис.1. Общая схема геометрического метода формирования структуры нейронной сети.

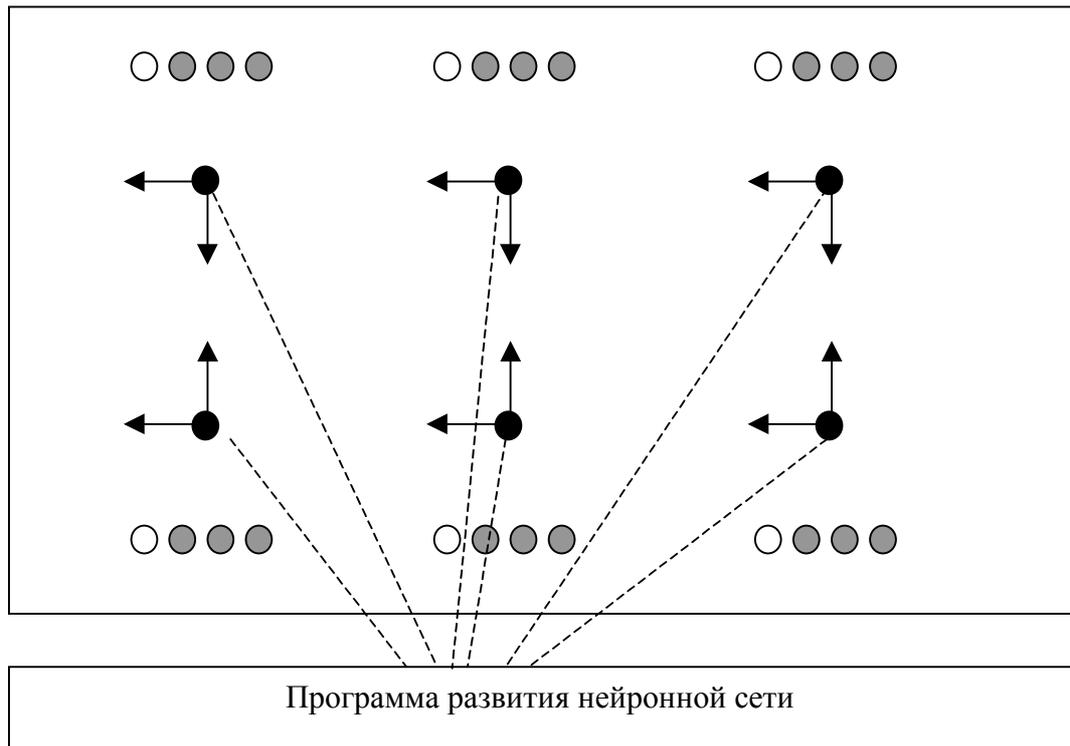
Эта схема иллюстрирует процесс формирования отдельного модуля нейронной сети. Аналогичным образом, к этому модулю могут добавляться новые модули. Причем сначала формируются модули нижних уровней, определяющие инстинктивное, рефлекторное поведение анимата (например, согласованное движение 6-ти ног при прямолинейном перемещении анимата). А затем формируются модули более высоких уровней (например, модуль управляющий остановкой и возобновлением прямолинейного движения), управляющие работой модулей нижних уровней.

2. Программа развития нейронной сети

Опишем принципы формирования нейронной сети на основе инструкций программы развития. Процесс формирования определяется как экспериментатором, конструирующим нейронную сеть, так и работой программ развития и эволюционной оптимизацией этих программ.

Первоначально экспериментатор выделяет двумерную область (например, прямоугольник). В этой области он задает расположение тех нейронов, которые заведомо необходимы (сенсорные и моторные нейроны). Затем задается множество затравочных нейронов, из которых будет развиваться нейронная сеть.

Пример области с расположенными на ней сенсорными, моторными и затравочными нейронами показан на рис.2.



- – сенсорные нейроны
- – мотонейроны трех разных типов, управляющие разными "мышцами" анимата
- – затравочные нейроны

Рис. 2. Схема расположения изначально задаваемых нейронов в прямоугольной двумерной области (по работе [6]). Симметрия схемы связана с тем, что она предназначена для формирования нейронной сети 6-ногого анимата. Каждый из затравочных нейронов имеет свою систему координат, которую он затем использует при формировании дочерних нервных клеток и связей с другими нейронами.

Экспериментатор также задает инструкции, из которых будут формироваться программы развития нейросетей, и некоторые синтаксические ограничения на то, как из этих инструкций можно формировать программы.

В работах Анимат-лаборатории была использована модель интегрирующего нейрона, согласно которой динамика мембранного потенциала i -й нервной клетки описывается уравнением:

$$\tau_i \frac{dm_i}{dt} = -m_i \sum w_{ij} x_j + I_i, \quad (1)$$

где $x_j = \{1 + \exp[-(m_j + B_j)]\}^{-1}$ – частота импульсов нейрона, B_j – случайный порог нейрона со средним значением b_j , τ_i – постоянная времени релаксации i -го нейрона, I_i – внешний вход i -го нейрона от определенного сенсора, w_{ij} – синаптический вес, характеризующий связь j -го от к i -му нейрону.

Преимущество данной модели нейрона состоит в том, что она позволяет аппроксимировать траекторию любой гладкой динамической системы.

Примеры инструкций, определяющих программы нейронов, представлены в Таблице 1.

Таблица 1. Инструкции – команды программы развития нейронной сети

DIVIDE (α, r)	Создать новый нейрон
GROW (α, r, w)	Создать соединение к другому нейрону
DRAW (α, r, w)	Создать соединение от другого нейрона
SETBIAS (b)	Изменить порог нейрона
SETTAU (τ)	Изменить постоянную времени релаксации нейрона
DIE	Удалить нейрон

Инструкции характеризуются параметрами, которые определяют, как именно будет развиваться нейронная сеть. Ниже будет рассмотрен пример, связанный с рис. 3.

Программа развития состоит из подпрограмм. Каждому затравочному нейрону соответствует своя подпрограмма.

Инструкции каждой из подпрограмм скомпонованы в граф, имеющий корневой узел, команда-инструкция которого применяется к затравочному нейрону. Работа подпрограммы начинается с того, что «затравочная» нервная клетка исполняет инструкцию корневого узла подпрограммы.

Одна из инструкций (DVIDE) соответствует делению клетки на материнскую и дочернюю. Формирование дочерней нервной клетки соответствует ветвлению графа подпрограммы; при этом после деления клетки (и соответствующего ветвления графа) инструкция левого узла относится к материнской нервной клетке, а инструкция правого узла – к дочерней клетке. Итак, инструкция DIVIDE соответствует узлу ветвления графа на две ветви. Все остальные инструкции не приводят к ветвлению графа подпрограммы.

Некоторые из команд-инструкций терминальные, после этих инструкций процесс развития той нервной клетки, к которой они применяются, останавливается.

В процессе эволюции программы испытывают мутации и рекомбинации. Рекомбинации представляют собой обмен подграфами подпрограмм – аналогично тому, как это осуществляется в генетическом программировании [9,10].

Пример применения инструкций показан на рис. 3.

- – развивающаяся нервная клетка
- – дочерняя клетка
- – другой нейрон

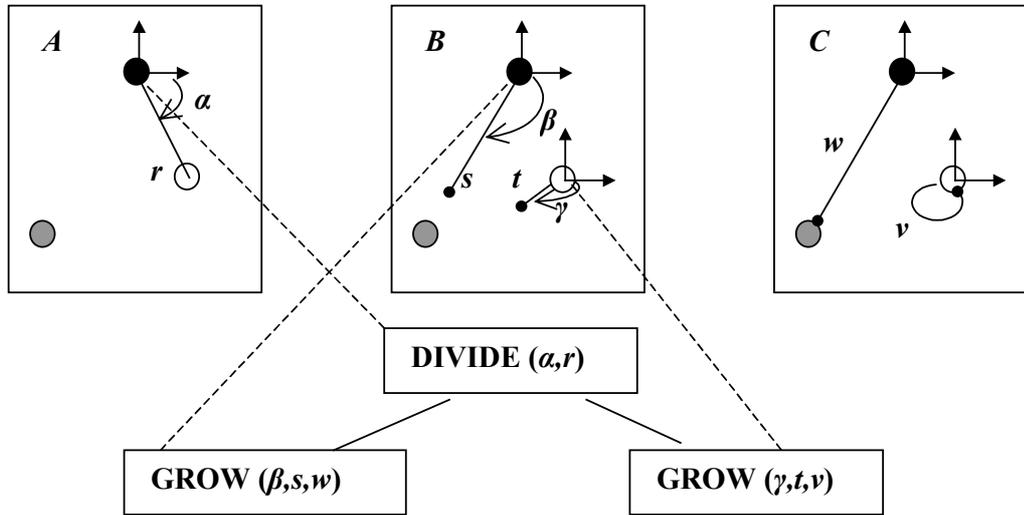


Рис. 3. Пример развития нейронной сети. Развитие происходит под управлением программы, представленной внизу рисунка. Программа содержит команду деления развивающейся нервной клетки **DIVIDE** (α, r) и команды формирования связей от материнской **GROW** (β, s, w) и дочерней **GROW** (γ, t, v) клеток. Первые два параметра в этих командах определяют углы и расстояния, в соответствии с которыми определяется положение дочерней клетки (для команды **DIVIDE**) и направление и длина синаптической связи (для команды **GROW**). Третий параметр команды **GROW** определяет величину синаптической связи.

Как видно по рисунку 3(A), сначала по команде **DIVIDE** (α, r) формируется дочерняя клетка на расстоянии r от материнской с "азимутом" α . Дочерняя клетка наследует локальную систему координат материнской клетки. Затем (рис. 3(B)) как материнская (по команде **GROW** (β, s, w)), так и дочерняя (по команде **GROW** (γ, t, v)) клетки формируют отростки длиной s и t , и под углом β и γ соответственно. Далее (рис. 3(C)) концы этих отростков подсоединяются к ближайшей из клеток и задаются веса соответствующих синаптических связей. На этом указанный блок программы полагается выполненным.

Итак, геном анимата представляет собой определенную программу, однозначно определяющую процесс формирования структуры и весов нейронной сети. Программа состоит из инструкций, которые задают правила расстановки нейронов в заданной геометрической области и формирования синаптических связей между нейронами.

3. Эволюционный алгоритм

Эволюционный алгоритм состоит в следующем. Поведение анимата оценивается в соответствии с эвристически задаваемой функцией приспособленности, производится отбор аниматов с большими приспособленностями, и наиболее приспособленные аниматы дают потомков. Конкретная схема отбора может быть различной. В работе [6] используется несколько запутанный выбор лучшей пары особей из некоторого множества, формирование потомка этой пары и замещение потомком худшей особи из двух случайно выбранных аниматов.

При формировании потомков применяется три генетических оператора: 1) совместимая рекомбинация подграфов программ выбранных родителей, 2) формирование новых случайных, но допустимых подграфов вместо старых, 3) и случайные мутации параметров, входящих в команды-инструкции.

В целом схема эволюции типична для генетического алгоритма, однако, необходимо наложение определенных условий на то, чтобы программы потомков не выходили за рамки ограничений, накладываемых на эти программы.

4. Принцип модульности. Примеры использования геометрического метода

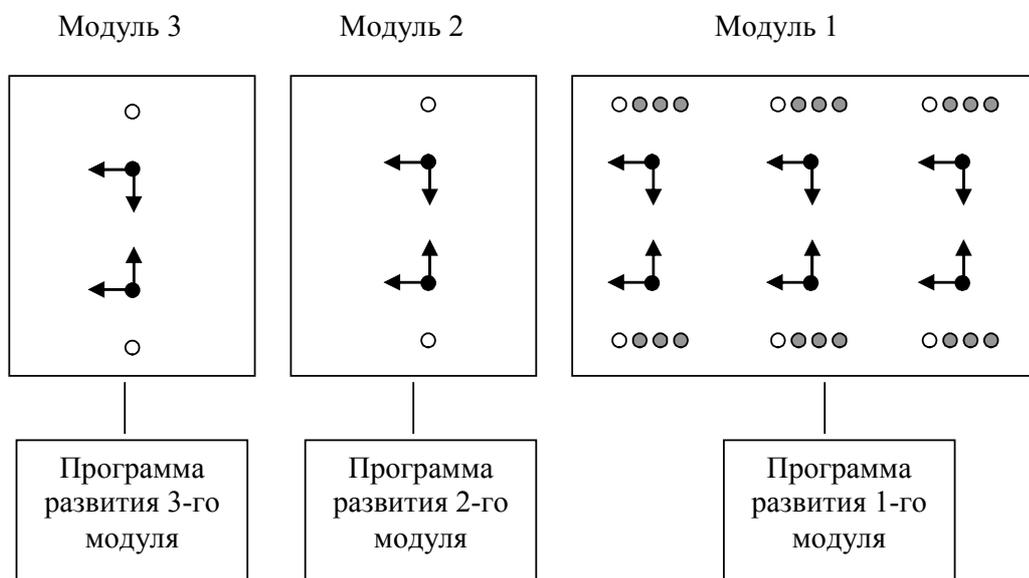
Принцип модульности подразумевает, что модули нейронной сети формируются последовательно – сначала один модуль, затем следующий. Например, в работе [6] сначала был эволюционно сформирован 1-й модуль, определяющий согласованное движение 6-ти ног при прямолинейном 1-мерном перемещении анимата, а затем 2-й модуль, управляющий остановкой и возобновлением прямолинейного движения.

В работе [7] была сформирована нейронная сеть, управляющая 2-мерным движением 6-ногого анимата. При этом нейронная сеть состояла из трех модулей: 1-й модуль управлял движением анимата, 2-й модуль контролировал работу первого модуля и обеспечивал перемещение анимата к заданной цели, а 3-й модуль обеспечивал перемещение в среде с препятствиями, и был предназначен для минимизации столкновений анимата с препятствиями.

Схема модульной структуры этого анимата представлена на рис.4. Модули формировались последовательно (сначала 1-й, потом 2-й, затем 3-й). При формировании 1-го модуля приспособленность анимата оценивалась по скорости его движения (чем больше скорость перемещения, тем выше приспособленность). При формировании 2-го и 3-го модулей 1-й модуль оставался неизменным, а формировались только связи от нейронов новых модулей к нейронам первого модуля. Приспособленность программ 2-го модуля оценивалась по способности анимата находить источник запаха. Приспособленность программ 3-го модуля оценивалась по способности анимата избегать препятствия, случайно разбросанные в области его движения. Программы формирования каждого из модулей были различными, более того, несколько различался синтаксис инструкций и графов, на основе которых формировались программы.

Сформированный 1-й модуль содержал 38 интернейронов (сформированных нейронов, обеспечивающих связи между сенсорными нейронами и мотонейронами) и

100 межнейронных соединений. 2-й модуль содержал 6 интернейронов и 22 межнейронных соединения. 3-й модуль содержал 2 интернейрона и 6 межнейронных соединений.



- – сенсорные нейроны
- – мотонейроны трех разных типов, управляющие разными "мышцами" анимата
- – затравочные нейроны

Рис.4. Схема формирования модульной структуры нейронной сети, управляющей движением 6-ногого анимата (по работе [7]). 1-й модуль предназначен для формирования устойчивого перемещения анимата по двумерной плоскости. 2-й модуль управляет движением, направленным на достижение целей (скажем, движение к источнику запаха пищи), 3-й модуль минимизирует столкновения с препятствиями.

В работах [6,7] геометрический метод формирования структуры нейронной сети был применен к задаче формирования системы управления аниматами, моделируемыми компьютерной программой, т.е. аниматы "жили" только в компьютере. В работе [8] этот метод был применен к реальному 6-ногому роботу SECT, который был обучен перемещаться по двумерной плоскости и избегать столкновения с препятствиями. Было продемонстрировано, что имитация поведения робота в компьютерных программах, разработанных в Анимат-лаборатории, согласуется с поведением реального робота.

5. Заключение

Анализ геометрического метода формирования структуры нейронной сети показывает, что этот метод достаточно универсален и может быть применен к широкому классу систем автономного адаптивного управления. Основные принципы этого метода сводятся к следующему.

1. Создается специальный язык команд-инструкций, на основе которого строятся программы развития (онтогенеза) структуры и параметров нейронной сети.
2. Программы оптимизируются эволюционным путем. Схема эволюционной оптимизации близка к таковой в генетическом программировании.
3. Сложная нейронная сеть строится по модульному принципу: сначала формируются модули нижних уровней управления, а затем модули верхних уровней иерархии управления. Программы формирования модулей разного уровня независимы друг от друга.

Данный метод прошел серьезную апробацию при создании нетривиальных систем управления реальными роботами.

Необходимо отметить, что использование этого метода подразумевает достаточно серьезную инженерную работу по конструированию языка команд-инструкций и конкретной схемы эволюционной оптимизации программ "онтогенеза" нейронных сетей.

6. Литература

1. W. B. Dress, "Darwinian optimization of synthetic neural systems," in Proceedings of the IEEE First International Conference on Neural Networks, SOS Printing, San Diego, CA, 1987.
2. A. Guha, S. Harp, and T. Samad, "Genetic synthesis of neural networks," Tech. Rep. CSD-88-14852-CC-1, Honeywell Corporate Systems Development Division, 1988.
3. D. Whitley, "Applying genetic algorithms to neural net learning," Tech. Rep. CS-88-128, Department of Computer Science, Colorado State University, 1988.
4. R. K. Belew, J. McInerney, and N. N. Schraudolph, "Evolving networks: Using the genetic algorithm with connectionist learning," Tech. Rep. CS90-174, CSE, UCSD, CA, June 1990.
5. G. F. Miller, P. M. Todd, and S. U. Hedge, "Designing neural networks using genetic algorithms," in Proceedings of the Third International Conference on Genetic Algorithms, Morgan Kaufmann, 1989.
6. Kodjabachian, J. and Meyer, J.A. (1998). Evolution and Development of Modular Control Architectures for 1-D Locomotion in Six-Legged Animats. *Connection Science*. 10, 211-237.
7. Kodjabachian, J. and Meyer, J.A. (1998). Evolution and Development of Neural Controllers for Locomotion, Gradient-Following, and Obstacle-Avoidance in Artificial Insects. *IEEE Transactions on Neural Networks*. 9, 796-812.
8. Filliat, D., Kodjabachian, J. and Meyer, J.A. (1999). Incremental Evolution of Neural Controllers for Navigation in a 6-legged Robot. In Sugisaka and Tanaka (Eds.). *Proceedings of the Fourth International Symposium on Artificial Life and Robotics*. Oita Univ. Press.

9. J. Koza, Genetic Programming: On the Programming of Computers by Means of Natural Selection. The MIT Press, 1992.
10. J. Koza, Genetic Programming II: Automatic Discovery of Reusable Subprograms. The MIT Press, 1994.