

14. Расчет t_{cp}^i и P_m^{i+1} .

15. Передача служебной информации агентам уровня $i - 1$.

16. Переход к п.1.

Разработанный алгоритм организует и оптимизирует распределенные вычисления методом принятия коллективного решения с использованием МС. Выполняется на фоне решения основной задачи. Время, затраченное на переконфигурацию структуры МС, из-за изменений в системе или поступления нового задания сводится к минимуму и оказывает относительно небольшое влияние на общее время решения задачи.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Лорин Г. Распределенные вычислительные системы. – М.: Радио и связь, 1984. – 296 с.
2. Таненбаум Э. Распределенные системы: принципы и парадигмы. – СПб: Питер, 2003. – 877 с.
3. Ховансков С.А., Мельник Э.В. Блушвили И.В. Метод организации распределенных вычислений в управляющих системах // Мехатроника, автоматизация, управление. «Новые технологии», №4, 2003.
4. Мельник Э.В., Иванов В.В., Погорелов К.В. Об одном подходе к организации поиска информации в Интернете // Известия ТРТУ, Тематический выпуск «Интеллектуальные и многопроцессорные системы». – Таганрог: Изд-во ТРТУ, №10, 2005.

М.В. Щербинина

ПРИМЕНЕНИЕ НЕЙРОСЕТЕВОЙ СТРУКТУРЫ МОДУЛЬНОГО ТИПА ДЛЯ РЕШЕНИЯ НАВИГАЦИОННОЙ ЗАДАЧИ*

Ставится задача определения координат летательного аппарата по измеренным расстояниям до опорных радиомаяков. Если рассматривается более трех радиомаяков, то аналитически задача не решается. Предложенные способы решения, основанные на методе наименьших квадратов, требуют больших вычислительных ресурсов. Поэтому предлагается использовать нейронные сети для решения поставленной задачи. Нейросетевые средства отличаются компактностью вычислений, при этом аппроксимируют желаемый результат с достаточно высокой точностью. В работе предлагается использовать модульную нейросетевую структуру для решения навигационной задачи.

Постановка задачи. Навигационная задача формулируется следующим образом (рис.1). В трёхмерном декартовом пространстве заданы координаты точек, в которых находятся опорные радиомаяки. Необходимо определить положение летательного аппарата (ЛА) по измеренным расстояниям D_i от ЛА до опорных радиомаяков ($i = \overline{1, n}$). Предполагается, что схема расположения маяков фиксирована, а ЛА находится внутри заданного куба, называемого областью решения. Необходимо с точностью не более чем 10 м оценить координаты летательного аппарата x , y и z по измерениям дальностей D_i до радиомаяков. Предполагается, что координаты радиомаяков заданы с погрешностями и измерения дальностей также производятся на фоне помех.

Поставленная прикладная задача относится к классу задач аппроксимации функций многих переменных. В роли функций выступают оцениваемые координаты

* Работа выполнена при финансовой поддержке РФФИ (проект № 05-08-01421).

ты x , y и z , а независимыми переменными являются D_i , $\overline{1, n}$. В дальнейшем для аппроксимируемых функций будет использоваться обозначение $F(v_1, v_2, \dots, v_n)$, а для независимых переменных – v_i , $i = \overline{1, n}$.

Обсуждение возможностей применения многослойного персептрона для решения навигационной задачи. Зависимости координат x , y и z от дальностей D_i , $\overline{1, n}$ являются гладкими нелинейными функциями. Как известно, гладкие нелинейные функции хорошо аппроксимируются с помощью искусственных нейронных сетей [1,2]. В рамках классической схемы для решения навигационной задачи можно использовать архитектуру – многослойный персептрон.

Качество построенного нейросетевого аппроксиматора оценивают по совокупности характеристик. Основными характеристиками являются сложность аппроксиматора, итоговая точность аппроксимации, вычислительные затраты на обучение персептрона. Под сложностью многослойного персептрона понимают сложность его архитектуры, т.е. число связей между нейронами. Избыточная сложность персептрона приводит к эффекту «переобучения» сети [1,2]. Итоговая точность аппроксимации представлена несколькими показателями ошибок, т.е. расхождений между желаемым и реальным выходом персептрона. Точность аппроксимации можно повысить, если взять достаточно много обучающих примеров, представляющих всю пространственную область аппроксимации. Вычислительные затраты ограничиваются техническими возможностями или установленным временем для решения задачи.

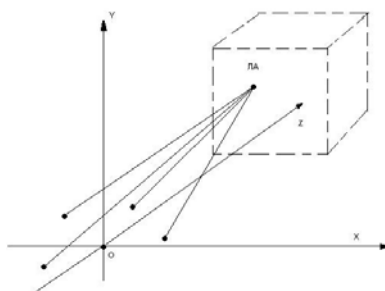


Рис.1. Иллюстрация к постановке навигационной задачи

Описанные характеристики оценки нейросетевого аппроксиматора взаимосвязаны. Возникают некоторые противоречия при построении персептрона, оптимизированного по всем характеристикам. Улучшение одной из них может привести к ухудшениям других. Поэтому нужно рассматривать некоторый компромисс по требованиям к каждой из перечисленных характеристик.

Для проведения эксперимента предполагалось, что используется 4 радиомаяка. Использовалась следующая область решения: x – от -5000 м до 5000 м, y – от 0 м до 4000 м и z – от -5000 м до 5000 м.

Проведенный анализ показал, что зависимости координат x , y и z от дальностей D_i не являются сложными. Чтобы обеспечить высокую точность аппроксимации, было создано порядка 500 000 примеров, равномерно распределенных в области решения. Из-за высокой вычислительной сложности заданную высокую точность (10 м) добиться не удалось. Поэтому исходную область решения было предложено «дробить» на фрагменты по координатам x , y и z , т.е. по пространству области аппроксимации.

На ограниченной области решения (x – от 0 м до 5000 м, y – от 2000 м до 4000 м и z – от 0 м до 5000 м; число точек около 50 000) был построен многослойный персептрон с числом нейронов по слоям 12:12:12 и сигмоидальными активационными характеристиками, который аппроксимировал координаты x , y и z со средней точностью 8 м.

Полученные результаты показали, что данную задачу невозможно решить в большой области решения с заданной точностью на многослойном персептроне, поэтому в качестве аппроксиматора уместно использовать модульную нейросетевую структуру.

Нейросетевая архитектура модульного типа. При решении задачи аппроксимации функции многих переменных $F(v_1, v_2, \dots, v_n)$ с помощью модульной нейросетевой структуры предполагается, что область изменения независимых переменных $v_i, i = \overline{1, n}$ разделена на локальные подобласти [3,4]. Подобласти также можно выделять и в области значений функций $F(v_1, v_2, \dots, v_n)$. Внутри каждой локальной области отдельно и независимо от других локальных областей может быть построена своя аппроксимация $\tilde{F}^{(j)}(v_1, v_2, \dots, v_n)$ функции ($j = \overline{1, K}$, где K – число локальных областей). Таким образом, прогноз значений $F(v_1, v_2, \dots, v_n)$ на всей области изменения независимых переменных формируется путем обобщения локальных прогнозов.

Модульная сеть (рис.2) состоит из классификатора и модулей, каждый из которых настраивается на локальную область.

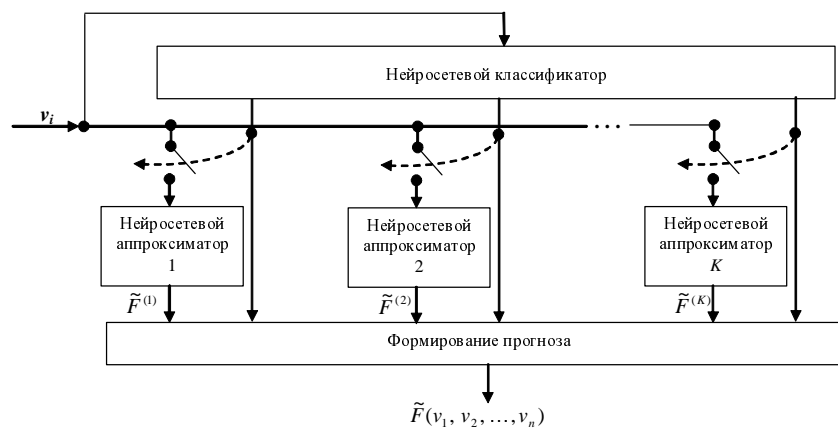


Рис.2. Общая схема модульной нейронной сети

Принадлежность точек к зоне действия модуля определяется классификатором, а сама локальная функциональная зависимость $\tilde{F}^{(j)}(v_1, v_2, \dots, v_n)$ – локальным аппроксиматором. Количество K таких модулей определяется числом локальных разнородных областей в описании сложной функциональной зависимости и заданной точностью аппроксимации. Нейросетевая структура включает в себя выходной модуль, который на основе работы локальных аппроксиматоров формирует итоговое значение аппроксимируемой зависимости. Для того чтобы не возникало резких изменений функции на границах действия модулей, локальные подобласти задаются с некоторым перекрытием. В этом случае обеспечивается гладкость функции $\tilde{F}(v_1, v_2, \dots, v_n)$.

В качестве классификатора могут быть использованы: многослойный персептрон, вероятностная нейронная сеть и пр. Локальный аппроксиматор может быть также реализован на многослойной нейронной сети или сети с радиально-базисными активационными характеристиками.

Каждый модуль описанной нейросетевой структуры обучается отдельно и независимо от других модулей. Это позволяет использовать для обучения каждого аппроксиматора значительно меньшее число обучающих примеров, нежели в сети, настраиваемой по всему пространству независимых переменных. Меньшая сложность функции в локальной области позволяет добиться более высокой точности для фиксированной архитектуры сети. А значит, не требуется использовать слишком сложные модули для решения задачи аппроксимации, что значительно сокращает время обучения. Кроме того, при обнаружении больших ошибок при обработке тестовых или обучающих примеров в определенной локальной области требуется перенастраивать только модуль (или несколько соседних модулей), действующий в данной локальной области.

Таким образом, нейросетевая архитектура модульного типа по сравнению с персептроном позволяет добиться более высокой точности, используя нейросетевые модули существенно меньшей сложности.

Подготовка обучающей выборки. Для решения навигационной задачи на модульной нейросетевой структуре предлагается разделить всю область значений функций x , y и z на 8 параллелепипедов – локальных областей. Четыре класса накрывали область высот от 100 м до 2000 м, а остальные - от 2000 м до 4000 м. Каждый класс располагался в проекции на плоскость (x, y) в одном из квадрантов. Постоянная ширина перекрытия классов составляла 500 м. Число радиомаяков – 4.

Принадлежность обучающего примера локальной области понимается, как принадлежность примера классу. Для обучения локальных классификаторов используются данные всей области значений функций x , y и z . Обучающий пример имеет вид, представленный в таблице.

Таблица
Общий вид обучающего примера нейросетевого классификатора

№ примера	D_1	D_2	D_3	D_4	C_1	C_2	...	C_8

В таблице под D_i , $i = \overline{1,4}$ понимаются дальности до радиомаяков, C_j – желаемые выходы классификатора – код примера, указывающий на принадлежность к классам ($j = \overline{1,8}$, $C_j = \{-1, +1\}$). Если пример принадлежит классу j , то j -й желаемый выход C_j равен $+1$, а остальные равны 1 . Возможна ситуация, когда пример лежит в перекрытии между классами k и l . В этом случае выходы C_k и C_l равны $+1$. Реальный выход классификатора C_j^{NN} принимает действительные значения в интервале $[-1; +1]$. Число примеров классификатора может быть существенно меньше, чем общее число обучающих примеров, так как обученный классификатор дает грубую оценку x , y и z . Если сработал j -й классификатор, то можно приближенно оценить x , y и z , как центр соответствующего параллелепипеда. Определить, принадлежность данного примера j -ому классу можно в первом приближении по знаку реального выхода C_j^{NN} .

При обучении классификатора возможна ситуация, когда желаемый выход $C_j = +1$, а реальный $C_j^{NN} < -\theta_j$, где θ_j – параметр, определяющий порог принадлежности примера классу j . Значение параметра θ_j для каждого класса определяет-

ся отдельно с учетом распределений значений выходов классификатора. Если $C_j^{NN} > \theta_j$, пример принадлежит классу j . Будем называть эту ситуацию ошибкой первого рода. Возможна и обратная ситуация: $C_j = -1$, а $C_j^{NN} \geq \theta_j$ – ошибка второго рода. Ошибки первого и второго рода наблюдаются, как правило, близко к границам классов. Подобные ошибки возникают вследствие обобщающего эффекта нейросетевого классификатора – примеры соседнего класса оказывают большое влияние на ошибочно классифицируемый пример и «затягивают» его в свой класс.

Поэтому формирование выборки для обучения локальных аппроксиматоров происходит после обучения и с учетом выходов классификатора. В обучающую выборку j -го аппроксиматора входят только те примеры, которые «распознаются» классификатором как принадлежащие j -ому классу.

Каждый локальный аппроксиматор строит в своем классе по значениям дальностей D_1, D_2, D_3 и D_4 прогноз значений координат \tilde{x}, \tilde{y} и \tilde{z} . Обучение нейросетевого аппроксиматора выполняется классическим методом обратного распространения ошибки [5].

Следует отметить, что на вход нейросетевой структуры следует подавать не непосредственно измеренные значения дальностей, а их преобразованные значения D_1, D_2, D_3 и D_4 . Нелинейное преобразование дальностей, как показал опыт, существенно сокращает время обучения классификатора и локальных аппроксиматоров, и позволяет использовать сети с меньшим числом слоев и нейронов.

Для формирования итогового прогноза \tilde{x}, \tilde{y} и \tilde{z} , кроме выходов локальных аппроксиматоров, используются выходы классификатора. Выходы классификатора предпологается использовать в роли весовых коэффициентов.

Обсуждение результатов нейросетевой аппроксимации. Для обучения классификатора модульной нейросетевой структуры в области решения было смоделировано около 45 000 примеров, располагающиеся в области решения в узлах решетки с ребром 200 м. В примеры вошли точки как не принадлежащие границам, так и принадлежащие границам классов. В качестве классификатора использовалась многослойный персептрон с сигмоидальными функциями активации.

Был проведен ряд экспериментов по построению классификаторов различных архитектур. Наиболее удачными по частоте ошибок первого и второго родов оказались персептроны с числом нейронов от 20 до 30 в двух скрытых слоях.

На рис.3 показаны гистограммы относительных частот для выхода C_2^{NN} классификатора при условии принадлежности выборочных примеров классу 2 ($C_2 = +1$). Гистограмма на рис.3,а построена по данным обучающей выборки, а на рис.3,б – по данным тестовой.

На рис.4,а представлены результаты статистической обработки выхода классификатора по данным обучающей выборки. Рис.4 содержит две построенные в общей системе координат гистограммы относительных частот, рассчитанные при условии, что входной вектор не принадлежит классу 2 (левая гистограмма, левая наклонная штриховка), и в противном случае (правая гистограмма, правая наклонная штриховка).

Гистограммы не перекрываются, так что можно выбрать порог θ , который определяет принадлежность примера классу. Аналогичные гистограммы приведены на рис.4,б для тестовой выборки.

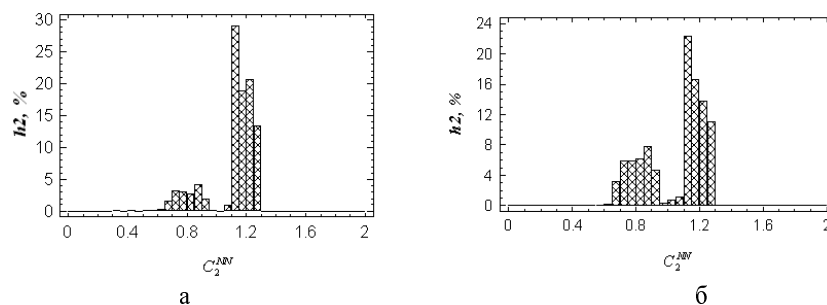


Рис.3. Гистограмма относительных частот для выхода 2 классификатора при условии принадлежности выборочных примеров классу 2

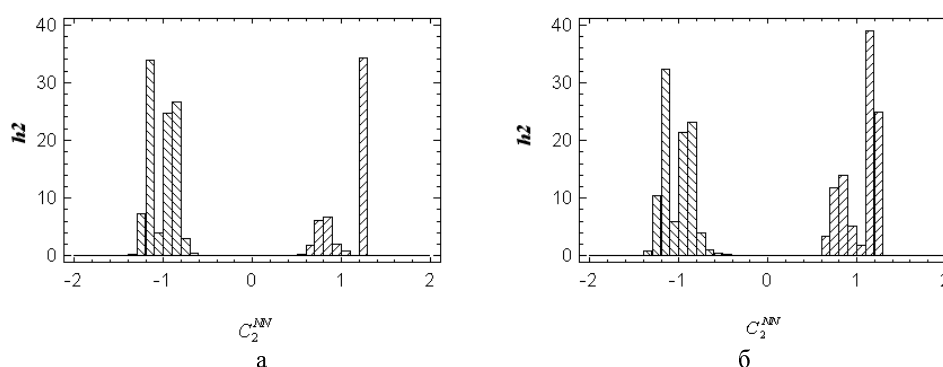


Рис.4. Гистограммы относительных частот для выхода 2 классификатора при условиях, что пример принадлежит классу C_2 и не принадлежит классу C_2

Анализ показывает, что обученный нейросетевой классификатор обладает свойством обобщения данных и позволяет также правильно классифицировать данные тестовой выборки.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Хайкин С. Нейронные сети. Полный курс. – М., С-Пб., Киев: Издательский Дом "Вильямс", 2005.
2. Осовский С. Нейронные сети для обработки информации. – М.: Финансы и статистика, 2004.
3. Оныкий Б.Н., Мишулина О.А., Похвалов Ю.Е., Харитонов В.С., Щербинина М.В., Построение аппроксимационной модели для критического теплового потока в трубах с использованием модульной нейронной сети // Сб. научных трудов «Нейроинформатика-2004», Ч. 1. – С. 159-166.
4. Мишулина О.А., Похвалов Ю.Е., Трофимов А.Г., Ускова А.А., Щербинина М.В., Реализация нейросетевой модели критического теплового потока в рамках пакета прикладных программ N-FONTO // Сб. научных трудов «Нейроинформатика-2005», Ч. 2, С. 172-182.
5. Rumelhart D. E., Hinton G. E., and Williams R. J. Learning Internal Representations by Error Propagation in Rumelhart. D. E. and McClelland J. L. // Parallel Distributed Processing: Explorations in the Microstructure of Cognition, MIT Press, 1986.