

задачи – корректное представление точек поискового пространства в виде дерева и вычисление пригодности.

Точкой поискового пространства является логическая формула. При представлении решения в виде дерева необходимо пользоваться следующим принципом: во-первых, все узлы дерева разбить на терминальные (конечные узлы у которых нет подузлов) и функциональные (узлы не являющиеся терминальными). Терминальные узлы являются элементами терминального множества, которое включает в себя фрагменты решения. Функциональные узлы являются элементами функционального множества, состоящего из операторов, которые из отдельных фрагментов – элементов терминального множества, формируют более крупные фрагменты и решение целиком.

Для выращивания логических формул в терминальное множество необходимо включить предпосылки и факты, описывающие задачу. Предпосылки и факты могут быть как статические, так и динамические. Соответственно, переменные, описывающие эти факты при чтении дерева должны обрабатываться по-разному.

В функциональное множество необходимо включить формулы, обрабатывающие факты из терминального множества, логические операторы, оператор вывода, и, кроме того, можно включить такие общезначимые формулы, как включение / исключения конъюнкта (дизъюнкта), если эти формулы, по мнению исследователя, могут быть полезными при логическом выводе.

При поиске решения необходимо производить оценку полученного дерева – пригодность. Как правило, пригодность определяется решаемой задачей. Однако можно дать общие рекомендации по вычислению пригодности в случае логического вывода. Цель поиска – найти тот вывод, который приводит к желаемым значениям логических переменных описывающих состояние системы. Кроме того, чем больше в дереве правильных промежуточных выводов, тем оно перспективнее. И, кроме того, необходимо наложить определенный штраф на длину дере-

ва, с целью предотвратить возникновение больших бесперспективных деревьев. В общем случае, расчет пригодности выглядит следующим образом:

$$E(x_i) = \begin{cases} 1, & x_i = \text{"истина"}, \\ 0, & x_i = \text{"ложь"}, \end{cases} \quad (1)$$

$$fitness = \frac{\sum_{i=1}^n (K_i \cdot (E(x_i))) + k \cdot N_{\square}}{K_{sh}}, \quad (2)$$

где n – количество переменных определяющих целевое состояние; x_i – переменная определяющая целевое состояние (если все n переменных примут значение «истина», то целевое состояние достигнуто). K_i – коэффициент определяющий «премию» за достижение i -й переменной целевого значения; N_{\square} – количество выполненных (логически допустимых) выводов в дереве; k – коэффициент определяющий «премию» за правильно выполненный вывод; K_{sh} – коэффициент определяющий штраф за длину дерева. K_{sh} целесообразно выбирать на порядок больше чем k так как в первую очередь нам важно достигнуть целевое состояние.

Таким образом, метод генетического программирования позволяет найти решение для достижения заданного целевого состояния, а предпосылки, входящие в полученное дерево решения, позволяют восстановить цепочку логических рассуждений.

Библиографические ссылки

1. URL: <http://ru.wikipedia.org>.
2. Goldberg D. E. *Genetic algorithms in search, optimization and machine learning* // Reading, MA : Addison-Wesley, 1989.
3. URL: <http://www.genetic-programming.com/gpanimatedtutorial.html>.

© Старовойтов С. А., Липинский Л. В., Семенкин Е. С., 2010

УДК 62.506.1

А. В. Стрельников
 Научный руководитель – А. В. Медведев
 Сибирский государственный аэрокосмический университет
 имени академика М. Ф. Решетнева, Красноярск

О НЕПАРАМЕТРИЧЕСКИХ МОДЕЛЯХ СТАТИЧЕСКИХ СИСТЕМ

Рассматривается задача идентификация статических процессов со статистической зависимостью между компонентами входной переменной.

Одной из основных проблем обработки экспериментальных данных является построение разнообразных математических моделей. Эта проблема возникает во всех сферах научных исследований и разработки. К настоящему моменту наиболее развита теория идентификации и моделирования в «узком»

смысле, т. е. параметрическая идентификация. Этот путь содержит два этапа: на первом этапе происходит выбор с точностью до параметров модели исследуемого процесса; а на втором – оценка параметров, входящих в модель, по экспериментальным данным. Измерение входных-выходных переменных

происходит со случайными погрешностями. Такой подход не всегда возможен, так как требует достаточной априорной информации о процессе, которая позволит определить параметрическую структуру модели. На практике такая информация не всегда доступна. В этом случае используют идентификацию в «широком» смысле, в которой отсутствует этап выбора структуры. Модель в этом случае строится на основании экспериментальных данных, но при этом требует некоторой априорной информации о качественных свойствах процесса (например, класс процесса).

Пусть процесс статичен и описывается следующим уравнением:

$$x(t) = f(u(t), \xi(t)), \quad (1)$$

где $u(t)$ – входная переменная процесса $u \in \Omega(u) \subset R^k$, а $x(t)$ – выходная переменная объекта $x \in \Omega(x) \subset R$, $\xi(t)$ – случайные возмущения. Область изменения переменных $u(t)$ и $x(t)$ назовем $\Omega(x, u)$.

На практике интервалы изменения переменных $u(t)$ и $x(t)$ всегда известны. Таким образом, пространство входных-выходных переменных $\Omega(x, u)$ может быть представлено без нарушения общности в виде гиперкуба. При построении моделей многомерных систем чаще всего возникают ситуации, когда компоненты векторов входных переменных статистически зависимы.

Оказывается, что область существования процесса ограничивается не только гиперкубом $\Omega(x, u)$, но и содержится в его некоторой подобласти $\Omega^H(x, u) \subset \Omega(x, u)$. Этот факт иллюстрируется на рис. 1.

Следует обратить внимание на существенное от-

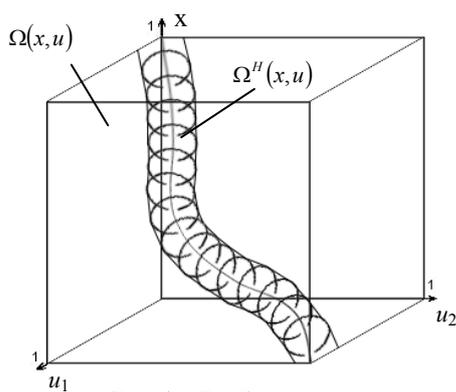


Рис. 1. «Трубчатая» структура

личие областей $\Omega(x, u)$ и $\Omega^H(x, u)$. В теории идентификации область $\Omega(x, u)$ всегда известна или задана. Область же протекания реального процесса $\Omega^H(x, u)$ никогда неизвестна.

Пусть модель процесса задана с точностью до вектора параметров $\alpha = (\alpha_1, \dots, \alpha_N)$:

$$\hat{x}(t) = \hat{f}(u, \alpha), \quad (2)$$

где \hat{x} – выход модели $\forall u \in \Omega(u) \subset R^k$. Поскольку идентификация в «узком» смысле состоит в оценивании параметров α по наблюдениям $x_i, u_i, i = \overline{1, S}$, то оценки α получаются соответствующими $\Omega_S^H(x, u)$, хотя последняя явно и не восстанавливается. Но при использовании модели (2) мы сталкиваемся с трудностями, состоящими в том, что для $(x, u) \notin \Omega^H(x, u)$, но $(x, u) \in \Omega(x, u)$ оценки \hat{x} могут существовать и вне допустимой области.

Смоделирован численный эксперимент, в котором была проведена прямая в пространстве и случайным образом сгенерировано пять выборок по 200 точек. Для каждой выборки строилась плоскость, проходящая через эту прямую. При этом среднеквадратичная ошибка не превышала десятой доли процента. Но, как показано на рис.2, каждая плоскость имеет существенно отличающиеся значения параметров.

Для избегания таких ситуаций предлагается использовать соответствующую модель:

$$\hat{x} = f(u, \beta) \cdot \Theta^H(u), \quad (3)$$

где $\Theta^H(u)$ – индикаторная функция. Ее оценка может иметь вид

$$\Theta^H(u) = \sum_{i=1}^s \prod_{k=1}^n \Phi\left(\left(c_s^k\right)^{-1} \left(u^k - u_i^k\right)\right), \quad (4)$$

где $\Phi(\cdot)$ – финитная колоколообразная функция, а c_s^k – коэффициенты размытости, удовлетворяющие условиям сходимости [1].

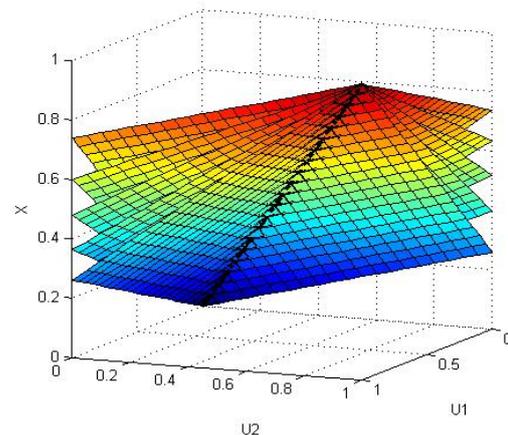


Рис. 2. Обычная идентификация

В задачах непараметрической идентификации статических и динамических объектов модели «трубчатой» структуры получаются естественным образом, поскольку непараметрические стохастические аппроксимации относятся к классу локальных.

В этом случае непараметрические модели уже соответствуют форме (3).

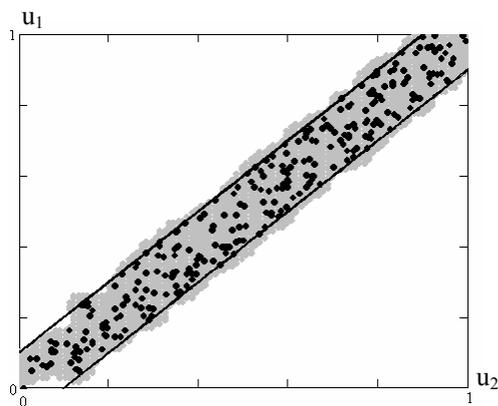


Рис. 3. Пространство входной переменной

Результат работы индикаторной функции представлен на рис. 3. Черными линиями показана реальная область протекания процесса, а серым цветом – ее оценка.

Библиографическая ссылка

1. Медведев А. В. Анализ данных в задаче идентификации // Компьютерный анализ данных и моделирование : сб. науч. статей Междунар. конф. Т. 2. Минск, 1995. С. 201–206.

© Стрельников А. В., Медведев А. В., 2010

УДК 621.314.2

Д. А. Токмин

Научный руководитель – Е. С. Семенкин

Сибирский государственный аэрокосмический университет
имени академика М. Ф. Решетнева, Красноярск

РАЗРАБОТКА И ИССЛЕДОВАНИЕ ЭВОЛЮЦИОННОЙ НЕЙРОННОЙ СЕТИ

Проводится анализ эффективности генетического алгоритма, генерирующего систему нейронных сетей с произвольной структурой для решения задачи прогнозирования.

Генетический алгоритм [1] – это эвристический алгоритм поиска, используемый для решения задач оптимизации путём случайного подбора, комбинирования и вариации искомым параметров с использованием механизмов, напоминающих биологическую эволюцию.

Искусственные нейронные сети [2] – математические модели, а также их программные или аппаратные реализации, построенные по принципу организации и функционирования биологических нейронных сетей – сетей нервных клеток живого организма. Это понятие возникло при изучении процессов, протекающих в мозге, и при попытке смоделировать эти процессы.

Для решения задачи прогнозирования был создан генетический алгоритм, оптимизирующий структуру нейронной сети. Настройка весов и параметра, определяющего наличие связи между нейронами на соседних слоях, производилась запуском отдельного генетического алгоритма. Для минимизации ошибки во время работы алгоритма использовался подход, заключающийся в генерации сразу нескольких нейронных сетей, каждая из которых настраивается отдельно и в качестве конечного выхода системы берется усредненное значение выходов созданных нейронных сетей. Входы нейронной сети перед запуском нормировались.

Объект прогнозирования – гидротурбина. Имеется вектор входных воздействий на объект \bar{x} , размерность которого $n = 11$, вектор выходных пара-

метров – значения вибрации снятые с различных точек на объекте – \bar{y} , размерность которого $m = 12$. Также есть обучающая выборка (\bar{x}_i, \bar{y}_i) , где \bar{y}_i , соответствующие набору входов \bar{x}_i , выходы, $i = \overline{1, 1000}$. Функциональная зависимость неизвестна. Необходимо сделать прогноз выходных параметров системы, для возможных входных воздействий.

В результате многократных запусков были получены результаты, свидетельствующие о хорошей работоспособности алгоритма. Средняя процентная ошибка по всем выходам системы нейронных сетей равна 7,4 %.

Использование нескольких нейронных сетей позволило уменьшить разброс ошибок на разных выходах, таким образом, ошибка настройки сети по каждому выходу стремится к средней ошибке по всем выходам, что положительно сказывается на качестве прогнозирования.

Библиографические ссылки

1. Goldberg D. E. Genetic algorithms in search, optimization and machine learning. MA : Addison-Wesley, 1989.
2. Уоссермен Ф. Нейрокомпьютерная техника. Теория и практика. М. : Мир, 1984.

© Токмин Д. А., Семенкин Е. С., 2010