

УДК 519.711

# КОНСТРУИРОВАНИЕ ИНТЕЛЛЕКТУАЛЬНЫХ РЕГУЛЯТОРОВ

**М. В. Бураков,**

канд. техн. наук, доцент

**А. С. Коновалов,**

д-р техн. наук, профессор

Санкт-Петербургский государственный университет  
аэрокосмического приборостроения (ГУАП)

В статье рассматриваются современные технологии интеллектуального управления динамическими объектами. Описывается методика синтеза регуляторов с базами знаний, опирающаяся на применение имитационного моделирования и генетического алгоритма. Приводится алгоритм нечеткого моделирования нелинейных динамических объектов. Обосновывается методика конструирования интеллектуальных регуляторов как сообщества конкурирующих интеллектуальных агентов.

The purpose of this paper is to present a modern technologies of intellectual control of dynamic objects. The technique of synthesis of regulators with knowledge bases basing on application of simulation modeling and genetic algorithm is described. The algorithm of fuzzy modelling of nonlinear dynamic objects is resulted. The technique of designing of intellectual regulators as communities of the competing intellectual agents is proved.

## Введение

Родовым признаком интеллектуальных регуляторов (ИР) является присутствие базы знаний (БЗ) о процессе управления объектом. Хотя знания могут иметь разную форму представления, для человека наиболее естественным является использование продукционных правил. Однако экспертного опыта обычно бывает недостаточно для подробного описания всей БЗ. Поэтому ИР конструируются в результате процесса самоорганизации, в ходе которого большую роль играет имитационное моделирование. Рис. 1 иллюстрирует общую схему обучения ИР, где  $Y(t)$  и  $Y^*(t)$  – реальный и желаемый выход модели;  $\varepsilon(t)$  – ошибка управления;  $U(t)$  – сигнал управления;  $G(t)$  – тестовый сигнал.

Для успешного синтеза ИР нужно обладать адекватной имитационной моделью (ИМ), выбрать структуру регулятора и алгоритм коррекции его параметров.

Если объект хорошо изучен и понятны физические принципы его функционирования, то ИМ может быть построена на основании математического описания. Если же известны только входные и выходные соотношения объекта, то можно попытаться описать его структуру с помощью универсальных аппроксиматоров – нейронной сети (НС) или нечеткой системы.

Эволюционная методика синтеза ИР с помощью генетического алгоритма (ГА) предполагает поиск субоптимального решения на базе популяции конкурирующих альтернативных описаний БЗ. Как показано ниже, этот подход естественным образом может быть расширен для решения задачи конструирования ИМ сложного объекта управления, а также при разработ-

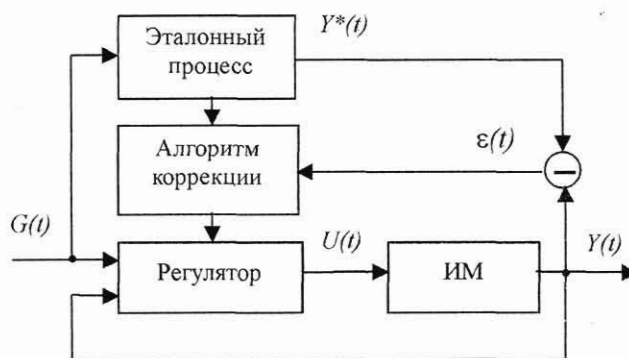
ке адаптивных систем управления, способных функционировать в условиях неполной информации о параметрах объекта и (или) внешней среды.

## Интеллектуальное управление

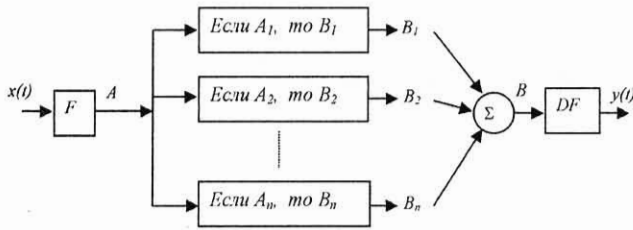
Рассмотрим наиболее популярные варианты представления знаний в интеллектуальных системах.

Ядром нечеткого логического регулятора (НЛР) является БЗ, состоящая из продукционных правил. Существуют два основных подхода к описанию нечетких правил: 1) модель *Mamdani* [1]; 2) *TSK*-модель (*Takagi, Sugeno, Kang*) [2]. Соответственно можно говорить о НЛР двух типов – *M*-типа и *TSK*-типа.

В НЛР *M*-типа консеквенты нечетких правил описываются как термы лингвистической переменной (ЛП) «Сигнал управления», а в НЛР *TSK*-типа консек-



■ Рис. 1. Обучение интеллектуального регулятора



■ Рис. 2. Система нечетких правил

вент представляет собой полиномиальную функцию входов.

Рассмотрим нечеткий регулятор *M*-типа. В простейшем варианте здесь каждое *i*-е правило имеет одну посылку  $A_i$  и одно заключение  $B_i$  (рис. 2, где *F* и *DF* – операции фаззификации и дефаззификации). Правила параллельно обрабатывают входную информацию  $x(t)$ , так что каждое правило порождает свой собственный элементарный выходной сигнал  $B_i$ . Общий выходной сигнал  $B$  является объединением (суммой) этих элементарных сигналов.

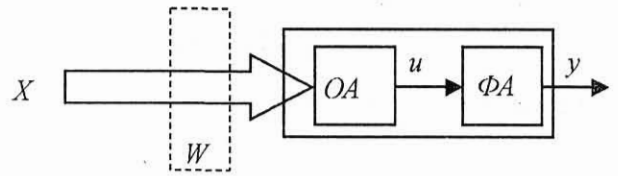
«Хорошая» система нечетких правил должна удовлетворять требованиям полноты и непротиворечивости. Полнота предполагает отсутствие во входном пространстве [области определения  $x(t)$ ] «белых пятен».

Непротиворечивость означает, что в базе не должно быть правил, которые имели бы при сходных посылках существенно различные заключения.

Главное достоинство системы нечетких правил заключается в плавности перехода от ситуации к ситуации. Кроме того, отдельное правило контролирует только некоторую область фазового пространства, так что закон управления в целом может быть нелинейным, и существует возможность независимого добавления или изменения отдельных правил (при соблюдении требований полноты и непротиворечивости).

Базовым элементом нейронного регулятора является искусственный нейрон (который в силу своей простоты имеет лишь грубое сходство с биологическим нейроном). Обобщенная модель нейрона показана на рис. 3 (где *OA* – операция агрегирования, которая сравнивает входной вектор и вектор весов, возвращая количественную оценку этого сходства  $u$ ; *ΦA* – функция активации).

Нейрон реализует нелинейное отображение



■ Рис. 3. Обобщенное описание нейрона

$$X \in R_n \rightarrow y \in R_1.$$

Векторы  $X$  и  $W$  обычно нормализуются. Для измерения сходства могут быть использованы различные подходы:

1) вычисление скалярного произведения векторов  $X$  и  $W$

$$u = XW = \sum_{i=1}^n x_i w_i;$$

2) вычисление эвклидова расстояния между векторами  $X$  и  $W$

$$u = \sqrt{(x_1 - w_1)^2 + (x_2 - w_2)^2 + \dots + (x_n - w_n)^2};$$

3) вычисление максимального отклонения

$$u = \max(|x_1 - w_1|, |x_2 - w_2|, \dots, |x_n - w_n|).$$

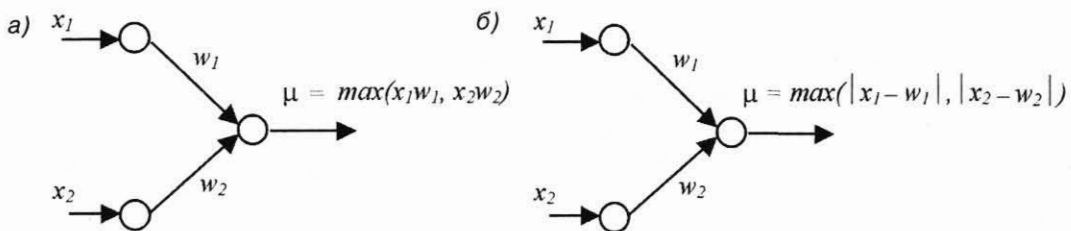
Активационная функция  $\Phi A$  выбирается, как правило, нелинейной (сигмоидная функция, линейная с насыщением и т. д.). Заметим, что нейрон с тремя входами, скалярным произведением в качестве измерения сходства и линейной активационной функцией соответствует обычному ПИД-регулятору.

Механизм «симбиоза» нечетких правил и НС достаточно прост в силу внутреннего сходства этих двух разных систем описания знаний [3].

Описание искусственного нейрона может быть выполнено с помощью понятий *T*-нормы и *S*-нормы (обобщенные операции *AND* и *OR*):

$$u = \sum_{i=1}^n [w_i T x_i].$$

При использовании в качестве *T*-нормы произведения, а в качестве *S*-нормы функции  $\max$  нейрону с двумя входами соответствует рис. 4, а.



■ Рис. 4. «Нечеткие» нейроны *OR* и *AND*

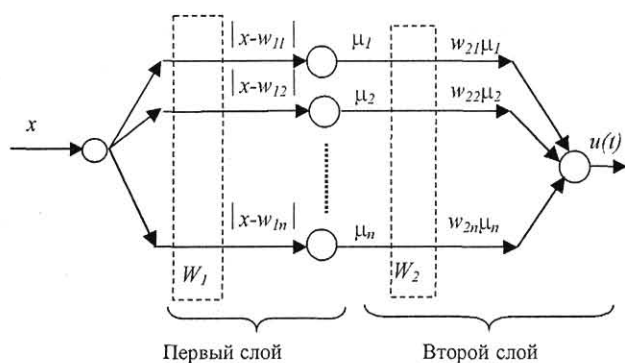


Рис. 5. Реализация системы правил нейронной сетью

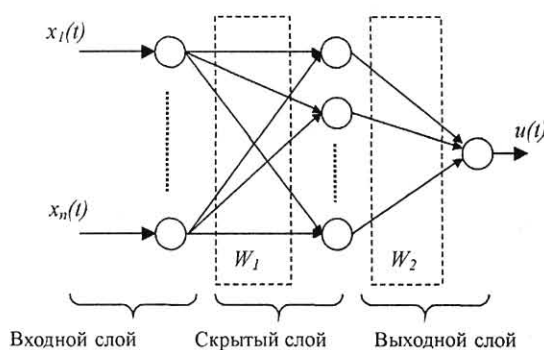


Рис. 6. НС для описания правил с n посылками

Очевидно, нейрону на рис. 4, а можно сопоставить логическую формулу:

$$(x_1 = w_1) \text{ OR } (x_2 = w_2).$$

Нейрон OR требует сходства хотя бы одного из входных сигналов с соответствующим весовым вектором для того, чтобы выходное значение  $u$  было близко к единице.

Однако посылки нечетких правил чаще связаны операцией AND. Нечеткий нейрон AND, наоборот, должен требовать, чтобы все входные сигналы (посылки) были близки соответствующим весовым векторам, поэтому здесь выгодно использовать критерий максимального отклонения (рис. 4, б).

При реализации нейрона AND можно использовать измерение декартова расстояния, но этот вариант не вполне отвечает представлениям о работе нечеткого правила, в котором слабое соответствие хотя бы одной посылки должно отражаться на общем соответствии посылок.

Таким образом, структура рис. 2 получает наиболее естественное воплощение при использовании НС, показанной на рис. 5.

НС на рис. 5 описывает систему правил с одной посылкой, здесь веса первого слоя  $w_{1i}$  отвечают за описание посылок, а веса второго слоя  $w_{2i}$  отвечают за описание заключений продукционных правил.

Рассмотрим стандартный механизм работы НЛР M-типа для объекта с  $n$  входов и одним выходом, реализующего отображение  $X \rightarrow R$ , где  $X \in R^n$ . Пусть входы обозначаются  $x_1, x_2, x_3, \dots, x_n$ .

Для каждого  $x_i$  определено  $m$  функций принадлежности  $\mu_{A_i}$ , каждая из которых реализует отображение входного сигнала в интервал  $[0, 1]$ . Они описывают  $A_i$  – термы ЛП, являющейся  $i$ -й посылкой нечеткого правила.

Так как всего  $n$  входов и каждый  $i$ -й вход имеет  $m_i$  функций принадлежности, то максимальное количество правил:

$$K = m_1 \times m_2 \times m_3 \times \dots \times m_n. \quad (1)$$

Декартово произведение  $A_1, A_2, \dots, A_n$  является нечетким множеством в пространстве  $X$  с функцией принадлежности, определяемой обычно в виде:

$$\begin{aligned} \mu_{A_1 \times A_2 \times \dots \times A_n}(x_1, x_2, \dots, x_n) = \\ = \min[\mu_{A_1}(x_1), \mu_{A_2}(x_2), \dots, \mu_{A_n}(x_n)]. \end{aligned} \quad (2)$$

Здесь каждая посылка представлена нечетким множеством  $A_i$ , а  $\mu_{A_i}$  описывает степень соответствия входа  $x_i$  терму  $A_i$ .

Нечеткое правило описывается следующим образом:

правило  $i$ : если  $(x_1$  есть  $A_1^i)$  и... и  $(x_n$  есть  $A_n^i)$ ,  
то  $u$  есть  $u_i$ ,

где  $A_j^i$  –  $j$ -я нечеткая посылка правила;  $u_i$  – одноточечное нечеткое множество.

Степень запуска  $i$ -го правила  $\mu_i$  получается по формуле (2).

Затем выходные сигналы всех правил комбинируются с помощью стратегии дефаззификации. В настоящее время предложены десятки методов дефаззификации [4], но обычно используется следующая формула (где  $N$  – количество правил):

$$u = \frac{\sum_{i=1}^N \mu_i \cdot a_i}{\sum_{i=1}^N \mu_i}. \quad (3)$$

В случае, когда правила продукционной системы имеют  $n$  посылок, эквивалентная НС преобразуется к виду, показанному на рис. 6.

Активационные функции нейронов первого слоя должны выдавать сигнал, величина которого обратно пропорциональна максимальному отклонению посылки правила, поэтому в качестве активационной функции здесь можно использовать вариант, представленный на рис. 7.

Нейрон второго слоя должен решать задачу дефаззификации, поэтому его активационная функция должна реализовывать формулу (3).

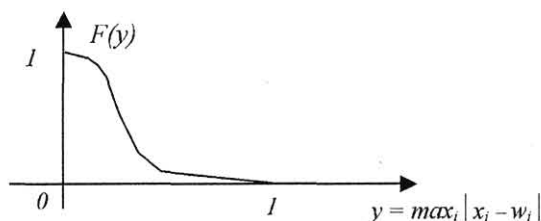


Рис. 7. Активационная функция нейрона первого слоя

Таким образом, двухслойная НС может использоваться для описания системы продукционных правил – наиболее простого и удобного способа представления знаний. Однако эффективное использование НС как основы ИР требует решения ряда проблем, в числе которых:

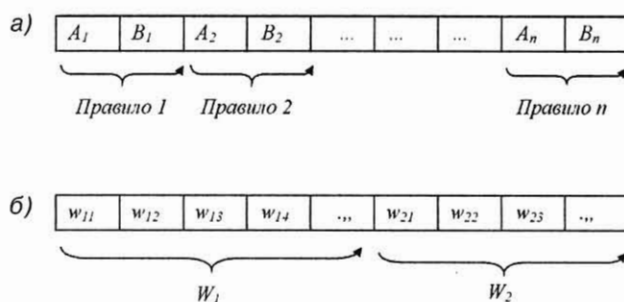
- определение необходимого количества нечетких правил (т. е. структуры НС); некоторые подходы к решению этой проблемы изложены в [5] и требуют анализа результатов имитационного моделирования;
- выбор параметров НС; эта задача решается в рамках структуры, показанной на рис. 1;
- проблема обеспечения адаптивности ИР; обученная НС трудно поддается оперативной коррекции – процесс обучения должен быть выполнен заново при изменении задачи управления или параметров объекта; иначе говоря, требуется решение известной «проблемы стабильности-пластичности» [6] НС.

### Эволюционный синтез регуляторов

БЗ определяет свойства интеллектуального регулятора. Формирование «хорошей» базы правил НЛР, равно как и настройка НС, являются сложными задачами, для решения которых требуется сочетать имитационное компьютерное моделирование и поисковые процедуры в пространстве параметров с целью минимизации заданного критерия качества. Градиентные методы поиска эффективны для унимодальных целевых функций и при малом пространстве поиска. Если же функция мультимодальная и (или) разрывная, то применяют методы случайного поиска, развитием которых является генетический алгоритм (ГА) [7, 8]. Его главное отличие от методов случайного поиска заключается в том, что он накапливает информацию о прошлых полученных решениях.

Генетический алгоритм моделирует естественный отбор – процесс выживания и воспроизведения организмов, наиболее приспособленных к условиям среды, и гибели в ходе эволюции неприспособленных. Резкие, внезапные мутации считаются решающим фактором эволюции, сразу ведущим к возникновению нового вида. Возникают мутации под воздействием на организм внешней среды.

Основой эволюционных процессов в органическом мире служит популяция – большая совокупность осо-



■ Рис. 8. Способы описания хромосом

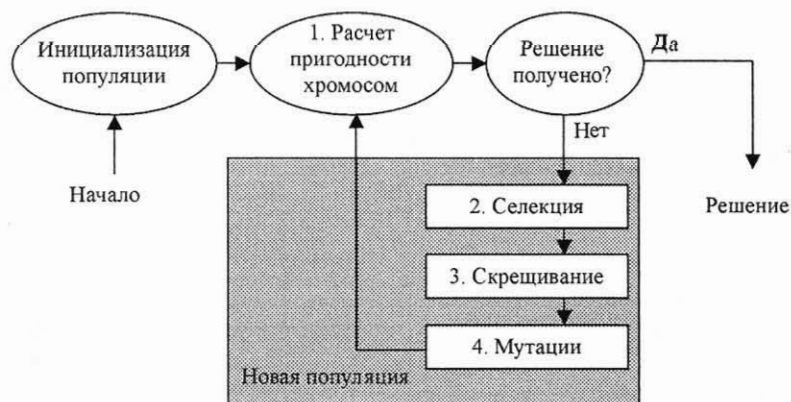
бей, каждая из которых эволюционирует самостоятельно, подчиняясь генетическим факторам.

Каждая особь популяции является обладателем уникального набора признаков, носителем которого являются хромосомы, состоящие из генов. Гены – это материальные структуры, ответственные за формирование признаков организма. Генотип (геном) – совокупность всех генов организма. Фенотип – совокупность всех признаков организма.

При конструировании ИР признаками являются параметры регулятора. Коды этих параметров образуют хромосомы, которые в процессе итераций тестируются и видоизменяются.

Для кодирования параметров могут использоваться вещественные числа или двоичный алфавит: {0, 1}. Цепочка битов длиной  $m$  может рассматриваться как хромосома, в которой отдельные позиции (биты) выступают в качестве генов. Хромосома для структур, показанных на рис. 2 и б, будут иметь соответственно следующий вид, показанный на рис. 8, а и б.

В общем случае, если длина цепочки равна  $m$  битов, то максимальный размер популяции оказывается  $2^m$ . При использовании популяции такого размера достаточно протестировать все решения и выбрать лучшее из них. Однако проблема заключается в том, что значение  $2^m$  в реальных задачах очень велико, а процедура тестирования хромосомы занимает вполне определенное время, поэтому реальный допустимый размер популяции  $N \ll 2^m$ .



■ Рис. 9. Общая структура генетического алгоритма

Работа ГА управляется тремя генетическими операторами: селекция, скрещивание, мутация (рис. 9).

Первоначальная генерация популяции осуществляется случайным образом в исследуемой области решений. После этого начинается циклическая работа ГА. На первом шаге происходит тестирование всех по очереди хромосом популяции, после чего каждая хромосома получает фиксированный на данном шаге признак – относительную пригодность (ОП) (*fitness function*). В рамках структуры рис. 1 ОП описывает расхождение между желаемым (эталонным) процессом на выходе системы и реальным выходом при подаче на вход тестирующего сигнала конечной длительности.

Если рассмотреть  $n$  моментов времени, то получается  $n$  обучающих пар:  $y$  и  $z$ , где  $y$  – реальный выход системы,  $z$  – желаемый выход. Тогда ошибка может быть определена в виде:

$$\varepsilon = \sum_{i=1}^n (y_i - z_i)^2 \quad \text{или} \quad \varepsilon = \sum_{i=1}^n |y_i - z_i|, \quad (4)$$

и ОП рассчитывается по формуле

$$ОП = \frac{1}{1 + \varepsilon}.$$

Таким образом, для получения ОП хромосомы требуется выполнить моделирование всего переходного процесса в системе. Операция оценивания пригодности является основополагающей для правильной селекции, а значит и для эффективной работы ГА. Однако простые формулы (4) не всегда пригодны для расчета ОП. В общем случае задача присвоения ОП фактически является задачей распознавания образов [9].

Селекция (отбор) является основной генетической операцией. Здесь возможны два разных подхода: либо при селекции отбираются отдельные признаки хромосом [10, 11], либо в результате селекции отбираются целиком хромосомы, пригодность которых превышает некоторый порог. На основании селекции происходит генерация новой популяции (популяции потомков).

Скрещивание выполняется с целью комбинирования и смешения признаков в популяции потомков. Мутация небольшого количества генов призвана сообщить потомкам новые признаки, которые могли отсутствовать в родительской популяции [12 и др.].

Схема ГА, показанная на рис. 8, предполагает, что количество потомков равно количеству родителей, и после выполнения мутации новая популяция полностью сформирована, так что может начинаться новый цикл развития. Однако при этом существует опасность потери лучших хромосом, поскольку все потомки могут оказаться хуже своих родителей. Для избежания этой опасности часто используются следующие простые приемы:

- выделяется элита популяции (около 10 %), которая попадает в новую популяцию без изменения;
- выделяются наихудшие хромосомы (также около 10 %), которые замещаются хромосомами, сгенерированными случайным образом.

Остальные хромосомы подвергаются генетическим операциям.

Для анализа текущего состояния популяции и более «справедливого» назначения ОП хромосом могут быть введены следующие критерии:

- селективное давление – отношение ОП наилучшей хромосомы к средней ОП всей популяции;
- селективная распространенность – количество возможных потомков одной хромосомы.

Существуют два подхода к присвоению ОП хромосом: пропорциональное и ранжированное присвоение.

Если численная оценка, которая связывается с каждой хромосомой, используется без изменения в операции селекции, то говорят о пропорциональном присвоении ОП.

Пропорциональное присвоение ОП может вызывать такие негативные явления, как стагнация при низком селективном давлении или преждевременное сужение зоны поиска (попадание в локальный минимум).

Ранжированное присвоение ОП предполагает дополнительную сортировку хромосом популяции и пересчет ОП, полученного на основании целевой функции, так что новое значение ОП зависит только от положения (ранга) хромосомы. Это позволяет контролировать селективное давление и ограничить количество потомков одной хромосомы.

Пусть хромосомы популяции отсортированы так, что лучшая хромосома имеет номер  $N$ , а худшая – номер 1 (где  $N$  – размер популяции). Тогда линейное ранжирование всех хромосом можно выполнить с помощью формулы

$$ОП(i) = 2 - CD + 2(CD - 1) \frac{(i - 1)}{(N - 1)},$$

где  $CD$  – селективное давление;  $ОП$  – относительная пригодность;  $i$  – номер хромосомы.

Могут быть также предложены формулы для нелинейного ранжирования, с помощью которых можно повысить селективное давление.

Рекомендации по использованию ГА для синтеза нейронных и нечетких регуляторов изложены в работе [9]. На практике выгодно использовать двух-трехслойные НС, поскольку иначе усложняется процесс обучения и тестирования. Однако сложность структуры регулятора зависит, естественно, от сложности управляемого объекта, и простая НС не способна реализовать сложную управляющую функцию. В работах [13, 14] обосновывается подход, связанный с генетическим синтезом множества относительно простых регуляторов для разных областей фазового пространства сложного объекта управления.

### Нечеткое моделирование

Задача кластеризации фазового пространства сложного объекта может выполняться для двух целей:

- 1) при моделировании объекта – с целью сопоставления каждому кластеру некоторой простой модели для описания выходного сигнала;
- 2) при конструировании регулятора – с целью поиска сигнала управления, который должен соответствовать каждому кластеру.

Рассмотрим первую задачу. Для ее решения может быть использован регулятор TSK-типа [15, 16].

Общий подход к моделированию сложных нелинейных систем заключается в линеаризации системы в заданной рабочей точке. Это может быть полезно для локального анализа, но не позволяет выполнять глобальный анализ поведения системы. TSK-модель позволяет агрегировать множество линеаризованных моделей для удобной аппроксимации сложной нелинейной системы.

Построение TSK-модели требует решения двух задач – структурной и параметрической идентификации.

Структурная идентификация предполагает следующие шаги:

1) определение необходимого количества правил «Если – То»;

2) разбиение входного пространства с помощью множества функций принадлежности.

Параметрическая идентификация заключается в уточнении параметров функций принадлежности посылок и заключений нечетких правил.

Рассмотрим нелинейную систему с  $n$  входами и одним выходом:

$$y = f(x_1, x_2, \dots, x_n).$$

Пусть известны  $m$  рабочих точек  $x_1^j, x_2^j, \dots, x_n^j$ ,  $j = \overline{1, m}$ , с помощью которых описывается  $m$  линеаризованных подсистем:

$$y^j = f^j(x_1, x_2, \dots, x_n).$$

TSK-модель представляется в виде набора правил:

$$R^j: \text{if } (x_1 \text{ есть } A_1^j) \text{ и } (x_2 \text{ есть } A_2^j) \\ \text{и ... то } y^j = f^j(x_1, x_2, \dots, x_n)$$

Выход TSK-модели описывается в виде

$$f_{fuzzy} = y = \sum_{j=1}^m b^j y^j,$$

где  $b^j = \frac{\tau^j}{\sum_{j=1}^m \tau^j}$ ;  $\tau^j = \prod_{i=1}^n A_i^j(x_i)$ ,  $\tau^j$  – степень запуска  $j$ -го

правила;  $y^j$  – линейная функция в  $j$ -й рабочей точке.

Функции принадлежности  $i$ -й входной переменной вокруг  $j$ -й рабочей точки описываются формулой

$$A_i^j(x_i) = \exp\left(-\frac{(x_i - \alpha_i^j)^2}{(\beta_i^j)^2}\right),$$

где  $\alpha$  и  $\beta$  – заданные параметры.

Разбиение входного пространства на области, относящиеся к рабочим точкам, и связывание этих областей в рамках TSK-модели позволяет получить удобную в использовании модель нелинейной системы. Мощность такого подхода зависит от того, позволяет ли сложная система выполнить декомпозицию на совокупность более простых подсистем, каждая из которых может быть описана линейной моделью.

При обучении должны быть заданы пары входных и выходных данных  $(x^k, y^k)$ ,  $x^k \in X \subset R^n$ ,  $y^k \in Y \subset R$ , которые сгенерированы неизвестной системой  $g(x)$ .

$$x = (x_1, x_2, \dots, x_n)^T \in X \subset R^n.$$

Задача обучения заключается в поиске нечеткой TSK-модели  $f_{fuzzy}(x^k)$ , такой, что для заданного порога ошибки  $\epsilon$  выполняется:

$$|g(x^k) - f_{fuzzy}(x^k)| = \left| g(x^k) - \sum_{j=1}^m \beta^j(x^k) f^j(x^k) \right| \leq \epsilon. \quad (5)$$

Необходимо использовать как можно больше данных, чтобы получить хорошую TSK-модель.

Обозначим заданное множество входных и выходных пар  $(x^k, y^k)$  как  $G^0$ .

Работа блока разбиения включает несколько шагов.

1. Определение линейной аппроксимирующей функции  $f_0$ . Здесь используется метод наименьших квадратов:

$$f_0(x) = B^T x;$$

$$B = (X^T X)^{-1} X^T Y,$$

где  $X[n \times m]$ ,  $Y[m \times 1]$ ,  $B[n \times 1]$ .

Вектор  $B[n \times 1]$  представляет собой параметрические оценки. Для того чтобы система была невырожденной, должно выполняться условие  $m \geq n$ .

2. Поиск точки максимума ошибки  $x^*$  для группы входных данных  $G^0$ , где максимизируется значение

$$|f_0(x^k) - y^k|, \quad k = \overline{1, m}.$$

Поскольку построение  $f_0$  базируется на усреднении данных, можно сделать вывод, что в точке  $x^*$  существует нелинейность.

3. Разделение групп данных на подгруппы.

После нахождения точки максимума ошибки

$$x^* = (x_1^*, x_2^*, \dots, x_n^*)^T$$

данные разделяются на две группы по всем входным переменным:

$$G_{i1}^1: (x^{k_{i1}}, y^{k_{i1}}), \quad k_{i1} = 1, \dots, c_{i1}, \quad \text{где } x_i > x_i^*;$$

$$G_{i2}^1: (x^{k_{i2}}, y^{k_{i2}}), \quad k_{i2} = 1, \dots, c_{i2}, \quad \text{где } x_i < x_i^*.$$

Здесь  $i = \overline{1, n}$  и  $n + 1 \leq c_{i1}, c_{i2} \leq m$  – размерности групп.

Число пар в каждой группе должно быть больше числа входных переменных.

После  $n$  разбиений мы будем иметь  $2n$  подгрупп данных:

$$G_{ij}^1: (x^{k_{ij}}, y^{k_{ij}}), \quad i = \overline{1, n}, \quad j = 1, 2, \quad k_{ij} = 1, \dots, c_{ij}. \quad (6)$$

4. Определение значимости входных переменных линейной аппроксимирующей функции  $f_{ij}$  (где  $i = \overline{1, n}, j = 1, 2$ ). Для этого выполняется расчет значения  $\rho_j$ :

$$\rho_i = \sqrt{\frac{\sum_{k=1}^m \left( \sum_{j=1}^2 f_{ij}(x^k) a_{ij}(x^k) - y^k \right)^2}{m}}$$

где переключательная функция описывается формулой

$$a_{ij}(x^k) = \begin{cases} 1, & x^k = x^{kj} \\ 0, & x^k \neq x^{kj} \end{cases}$$

Входная переменная  $x_i$  является наиболее значимым входом, если значение  $\rho_i$  для нее меньше, чем для всех остальных.

Разделение входного пространства на два подпространства относительно наиболее значимой входной переменной позволяет построить начальную TSK-модель, которая содержит два правила вида:

$$R^1: \text{if } x_i \text{ есть } A_i^1, \text{ то } y^1 = a_0^1 + a_1^1 x_1 + \dots + a_n^1 x_n;$$

$$R^2: \text{if } x_i \text{ есть } A_i^2, \text{ то } y^2 = a_0^2 + a_1^2 x_1 + \dots + a_n^2 x_n.$$

Очевидно, если эта модель достаточна для аппроксимации входной и выходной зависимости, то можно далее использовать ГА для оптимизации параметров этой модели. Дальнейшее разделение входного пространства не нужно, если выполняется условие (5), иначе необходимо дополнительное разделение входного пространства на два подпространства в соответствии с (6).

Если нет необходимости в разделении обеих групп входных данных, значит, нелинейность проявляется только в одной из них. Тогда дальнейшее разделение необходимо только для одной из групп данных.

В этом случае с каждой из групп данных  $G_{ij}$  ассоциируется величина

$$\rho_{ij} = \sqrt{\frac{\sum_{k_j=1}^{c_{ij}} \left( f_{ij}(x_{k_j}) - y^{k_j} \right)^2}{c_{ij}}}$$

где  $k_j = 1, \dots, c_{ij}$ , ( $c_{ij}$  – размерность  $G_{ij}$ ).

Таким образом, если с  $G_{ij}$  ассоциирована линейная функция  $f_{ij}$ , имеющая большое значение  $r_{ij}$ , то необходимо дальнейшее разделение пространства  $G_{ij}$ .

Оптимизация начальной TSK-модели может выполняться с помощью ГА. В ходе оптимизации корректируются параметры функций принадлежности или параметры линейных аппроксимирующих функций.

Заметим, что TSK-модель позволяет также аппроксимировать сложную управляющую функцию как набор локальных регуляторов, контролирующих разные области фазового пространства объекта. На границах областей действия регуляторов согласовываются. Приведенный алгоритм может использоваться для выбора структуры такого регулятора.

### Мультиагентная система управления

Традиционный подход к конструированию адаптивных систем заключается во введении в систему управления еще одного контура – контура адаптации, который обеспечивает коррекцию закона управления при неудовлетворительном качестве функционирования. Подобная идея может быть использована и по отношению к ИР [17 и др.]. Однако на этом пути встречаются значительные трудности:

- 1) ГА связан с большими вычислительными затратами и обычно не может использоваться для адаптации при управлении в реальном времени;
- 2) в НЛР оперативная коррекция одного правила управления затруднена из-за того, что в процессе дефазификации в каждый момент времени могут принимать участие множество правил;
- 3) в процессе адаптации трудно контролировать полноту и непротиворечивость управляющей БЗ, построенной на правилах;
- 4) сложная управляющая БЗ на основе НС плохо поддается коррекции в реальном времени, поскольку в ней трудно вычленить элементы, подлежащие изменению.

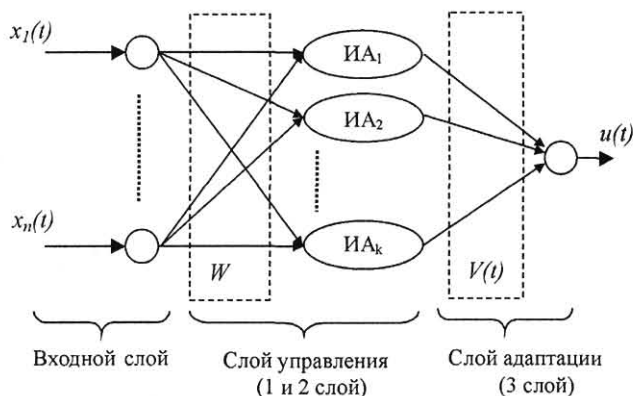
В связи с этим в последнее время большой интерес вызывает подход к созданию адаптивных ИР, связанный с использованием идеологии мультиагентных систем [18, 19 и др.].

Как было показано выше, двухслойная НС соответствует набору продукционных правил, описывающих БЗ нечеткого регулятора. Если же рассмотреть трехслойную НС, то ее можно уподобить (в рассматриваемом контексте) уже целому сообществу интеллектуальных агентов (ИА), решающих общую задачу (рис. 10).

Основная идея этого подхода заключается в том, что ИА, представляющие собой альтернативные описания БЗ, непрерывно получают информацию о поведении управляемого объекта. Каждый из ИА предлагает свой вариант управления в текущий момент времени (таким образом, ИА являются регуляторами-конкурентами). Все варианты оцениваются, так что на объект поступает наиболее предпочтительный вариант. С этой целью вектор весовых коэффициентов  $V$  должен изменяться во времени.

В зависимости от решаемой задачи в структуру, показанную на рис. 10, может вкладываться разное содержание. ИА могут описывать варианты стратегий управления в разных областях фазового пространства (ФП) системы, или описывать законы управления в зависимости от текущих параметров системы или внешней среды. В первом случае мультиагентная система (МАС) является обобщением TSK-регулятора, а во втором случае возникает адаптивная МАС (АМАС).

Рассмотрим задачи, которые могут быть решены в рамках МАС. Здесь происходит декомпозиция общей задачи управления на ряд подзадач, определенных



■ Рис. 10. Мультиагентная управляющая система

для подобластей ФП, в котором происходит движение управляемого объекта. Управление рассматривается как последовательность перемещений из одной области ФП в другую. Каждой области соответствует локальный регулятор. На границах областей действия регуляторов согласовываются [14]. Такой подход позволяет контролировать сложность и трудоемкость процесса обучения ИА. Если для какой-то области ФП не удастся добиться нужного качества управления, то эта область должна быть разбита на подобласти.

В рамках АМАС необходимо использовать набор ИА, БЗ которых синтезированы при существенно различных параметрах модели [20]. Адаптация заключается в активизации ИА, наиболее адекватного текущей ситуации на объекте, и одновременном подавлении неадекватных в текущий момент времени ИА. Таким образом, закон изменения  $V(t)$  является центральной проблемой при конструировании адаптивной МАС, для решения которой можно предложить два подхода – идентификационный и безидентификационный.

В первом случае структура МАС должна быть дополнена контуром идентификации, который на основании анализа входных и выходных сигналов пытается определить текущие параметры объекта и выбрать ИА, наиболее им соответствующий.

Во втором случае нужно анализировать только текущее качество управления и проводить поиск адекватного ИА. Для этих целей в работе [20] предлагается вычислять в реальном времени две функции – соответствия и полезности.

Функция соответствия должна описывать близость сигнала управления каждого ИА к сигналу управления, подаваемому на объект. Функция полезности оценивает текущее качество управления. При плохом качестве управления нужно подавать активные на предыдущем этапе регуляторы и активизировать ранее «заторможенные» регуляторы, предлагающие альтернативные варианты управления.

Предложенная структура может быть, в частности, использована в задаче разработки системы торможения колесами при посадке самолета [21]. Здесь большое значение имеет состояние взлетно-посадочной полосы, которое заранее неизвестно. Полоса может быть мокрой, сухой, полусухой и т. д. – всего может быть выделен ограниченный набор

состояний. Каждому состоянию полосы соответствует свой локальный регулятор. Стремительность процесса посадки затрудняет оперативную коррекцию правил или функций принадлежности. Использование АМАС позволяет оперировать набором конкурирующих ИА, так что в итоге включается в работу только один из них, адекватный текущей ситуации.

### Заклучение

Любая интеллектуальная система может рассматриваться как сообщество агентов, коллективные взаимодействия которых порождает общую реакцию системы. В НЛР в качестве простых агентов выступают отдельные правила управления, в нейронном регуляторе это отдельные нейроны. В мультиагентном интеллектуальном регуляторе в качестве агентов выступают целые БЗ, описывающие альтернативные стратегии управления.

Соответственно при конструировании нечеткого (или нейронечеткого) регулятора анализ экспериментальных данных имеет целью определение необходимого количества правил управления [5].

Если же конструируется интеллектуальная мультиагентная система, то сначала происходит анализ данных моделирования для определения областей фазового пространства, в которых могут функционировать локальные регуляторы (интеллектуальные агенты). На втором этапе может быть выполнен собственно синтез регуляторов. Такие регуляторы должны быть достаточно просты, чтобы не возникало проблем с их обучением и интерпретацией их поведения. Это позволяет контролировать сложность задачи синтеза системы управления в целом, оставляя возможность для контроля получаемых решений.

Описанный подход к синтезу интеллектуальных систем управления может быть перспективен при построении систем управления широким спектром динамических объектов и систем, в том числе – в авиационной технике [14, 21].



Литература

1. **Mamdani E. H.** Application of fuzzy algorithms for control of simple dynamic plant. – IEEE Proc., – 1974. – v.121. – № 12. – P. 1585–1588.
2. **Lee C. C.** Fuzzy logic in control systems: fuzzy logic controller – part 1 and 2. IEEE Transaction on System. – Man and Cybernetics. – 1990. – 20(2). – P. 404–435.
3. **Gupta M. M., Rao D. H.** On the principles of fuzzy neural networks // Fuzzy Sets and Systems. – Vol. 61. – 1984. – P. 1–18.
4. **Van Leekwijck W., Kerre E.E.** Defuzzification: criteria and classification // Fuzzy sets and systems 108 (1999). – P. 159–178.
5. **Бураков М. В., Коновалов А. С.** Нейронечеткие системы управления // Информационно-управляющие системы. – №1. – 2002 г. – С. 2–7.
6. **Уосермен Ф.** Нейрокомпьютерная техника: Теория и практика: Пер. с англ. – М.: Мир, 1992. – 240 с.
7. **Holland J.** Adaptation in Natural and Artificial Systems: An Introductory Analysis with application to Biology, Control and Artificial Intelligence. – University of Michigan Press, 1975.
8. **Goldberg D. E.** Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley, Reading, MA, 1989.
9. **Бураков М. В., Коновалов А. С.** Проектирование нейронных и нечетких регуляторов с помощью генетического алгоритма // Управление в условиях неопределенности. / Под ред. А. Е. Городецкого / СПб.: СПбГТУ, 2002. – 399 с.
10. **Захаров В. Н., Ульянов С. В.** Нечеткие модели интеллектуальных промышленных регуляторов и систем управления. Имитационное моделирование // Изв. РАН. Техническая кибернетика. – № 5. – 1994. – С. 168–204.
11. **Бураков М. В.** Синтез нейронного регулятора // Изв. АН/ Теория и системы управления. – 1999. – № 3. – С.140–145.
12. **Скурихин А. Н.** Генетические алгоритмы // Новости искусственного интеллекта. – № 4. – 1995. – С. 6–46.
13. **Бураков М. В., Коновалов А. С., Гиль А. В.** Алгоритмы синтеза интеллектуальных регуляторов // Тр. 2-й междунар. научно-практич. конф. «Информационные технологии в моделировании и управлении», 20–22 июня 2000 г. – СПб.: СПбГТУ. – С. 81–83.
14. **Бураков М. В.** Структура нейронечеткого регулятора // Изв. АН Теория и системы управления. – 2001. – № 6. – С. 160–165.
15. **T. Takagi, M. Sugeno.** Fuzzy identification of systems and its application to modeling and control // IEEE Trans. Systems Man Cybernet. – 15(1) 1985. P. 116–132.
16. **Baolin Wu, Xinghao Yu.** Fuzzy modelling and identification with genetic algorithm based learning // Fuzzy sets and systems 113 (2000). – P. 351–365.
17. **T. J. Procyk and E. H. Mamdani.** A linguistic self-organizing process controller, Automatica 15 (1979). – P. 15–30.
18. **Arvin Agah, Kazuo Tanie.** Fuzzy logic controller design utilizing multiple contending software agents // Fuzzy sets and Systems. – 106 (1999). – P. 121–130.
19. **Городецкий В. И.** Многоагентные системы: Современное состояние исследований и перспективы применения // Новости искусственного интеллекта. – 1996. – №1. – С. 44–59.
20. **Бураков М. В.** Механизм адаптации нечеткого регулятора // Изв. АН. Теория и системы управления. – 1998. – № 1. – С. 84–87.
21. **Бураков М. В., Коновалов А. С., Шумилов П. Е.** Нейронный регулятор в системе авиационной антиштурманской автоматики // V Всероссийская научно-технич. конф. «Нейроинформатика-2003». Москва, 29–31 января 2003 г.

ИЗДАТЕЛЬСТВО «ПОЛИТЕХНИКА» ПРЕДСТАВЛЯЕТ

**Куприянов М. С., Матюшкин Б. Д.**

**Цифровая обработка сигналов: процессоры, алгоритмы, средства проектирования. — 2-е изд., перераб. и доп. — СПб.: Политехника, 2002. — 592 с.: ил.**

Книга содержит три части. Первая часть «Процессоры цифровой обработки сигналов» посвящена архитектуре и особенностям организации DSP. Во второй части «Алгоритмы цифровой обработки сигналов» рассматриваются основы теории дискретных систем, методы анализа эффектов квантования сигналов при реализации алгоритмов обработки на DSP, базовые алгоритмы ЦОС и их реализация на DSP. Третья часть «Инструментальные средства проектирования систем ЦОС» содержит описание программных и аппаратных средств, используемых для решения задач проектирования и входящих в стартовый комплекс разработчика систем ЦОС. В приложении приведена система команд семейств DSP5600x и DSP5630x.

Книга рассчитана на инженерно-технических работников, занимающихся проектированием систем ЦОС, а также студентов соответствующих специальностей технических университетов.

