

УДК 519+61:681.3

ИНСТРУМЕНТЫ ИНТЕЛЛЕКТУАЛЬНОГО СИНЕРГЕТИЧЕСКОГО УПРАВЛЕНИЯ

Ю.А. Прокопчук

*Украинский государственный химико – технологический университет,**Институт технической механики НАНУ и НКАУ*

e-mail: itk3@ukr.net

Введение. Целью интеллектуального синергетического управления динамической системой является: 1) определение модели поведения системы (модель задает целевой аттрактор и параметры порядка разного уровня общности - $\{p/P\}$) и 2) реализация способа направленной самоорганизации, т.е. направленного движения вдоль желаемых инвариантных многообразий - аттракторов, к которым подстраиваются все другие переменные динамической системы. Отсюда вытекает идеология процессов обработки информации и управления в сложных нелинейных динамических системах. Необходимо, чтобы указанные процессы включали, по меньшей мере, две фазы: фазу расширения и фазу сжатия пространства состояний [1, 2, 3]. При этом в фазе расширения определяется модель поведения системы, формируется множество различных сценариев развития, задаются значения параметрам порядка. В фазе сжатия система сжимает область притяжения целевого аттрактора. На длительных интервалах наблюдения фазы расширения и сжатия будут чередоваться.

В рамках интеллектуального синергетического управления фазу расширения предлагается реализовывать на основе *модели знаний* и определенным образом организованного *мышления*. Инструментом мышления в рассматриваемом контексте является оператор замыкания (фаза расширения), который реализуется на основе модели предметной области. Модель ПрО представим в виде кортежа $\langle O, k \rangle$, где O – модель онтологии этой предметной области, а k – модель адекватной системы знаний. В развернутом виде модель знаний k представим следующим образом [4 - 6]:

$$k = \{f/\mu: k^1 \rightarrow k^2\} \cup P_k, \quad (1)$$

где f/μ – отображения, реализующие те или иные математические модели; μ – разные механизмы реализации отображений (со своей энергетикой и ресурсами); k^1 – входные данные задачи (описание информационной среды и задание); k^2 – выходные данные задачи; P_k – правила композиции схем задач, т.е. правила, описывающие способы объединения локальных задач.

Модель знаний (1) полностью отвечает синергетическим принципам: представляет собой взаимодействие (в процессе решения задачи) большого числа квантов знаний, представляемых отображениями; любой вычислительный процесс существенно нелинейный, так как является композицией различных отображений (детерминированных, нечетких, нейросетевых, агентных и т.д.); система открытая, так как активно осуществляется обмен с внешней средой (участие в консилиумах, непрерывный поиск информации в агентной среде и в Интернет); кванты знаний в общем случае имеют фрактальную структуру; метод предельных обобщений (МПО) описывает процессы горизонтальной и вертикальной самоорганизации, формируя минимальные избыточные модели знаний на любом уровне общности [7, 8]. «Параметры порядка» в знаниях – перечень тестов минимальной избыточной модели знаний с доменами максимального уровня общности. Таким образом, МПО создает степени свободы в описании ситуаций действительности, что, согласно теории синергетического управления, создает предпосылки для возникновения процессов самоорганизации.

Инструменты фрактальной организации знаний. Без ограничения общности предположим, что любую ситуацию действительности можно описать с помощью

множества элементарных тестов, которые будем обозначать $a, b, c, \{\tau\}$ и т.д. Примем также, что все тесты разбиваются на два подмножества: первичные тесты, т.е. тесты, значения которых должны быть заданы извне, и вычисляемые тесты, значения которых определяются на основе модели знаний (1). В соответствии с методом предельных обобщений [7] для каждого теста зададим ориентированный граф доменов, а для фиксации уровня общности будем использовать нотацию: a/A , где A – конкретный домен. Любой домен содержит полное множество значений теста на соответствующем уровне общности (гипотеза о замкнутости мира). Для обозначения результата теста будем использовать нотацию: $\underline{a}/A$. В рамках графа доменов любого теста задаются правила пересчета значений из каждого домена в другие – вышестоящие домены. Эти правила запишем так: $A \rightarrow A'$. Для правил пересчета выполняется закон транзитивности: если $A \rightarrow A', A' \rightarrow A''$, то $A \rightarrow A''$. Существует также базовый или начальный домен (обозначим его A_0) у которого нет предка, а именно: $\neg \exists A \neq A_0: A \rightarrow A_0$. Очевидно, значения из базового домена можно пересчитать в значения любого другого домена из ориентированного графа. Подобная иерархия доменов в пределе представляет собой нечто иное, как фрактал типа «канторовская пыль». Структура домена-фрактала задается фрактальным конфигуратором.

Пусть $W(\{b/B\})$ – некоторое многообразие на множестве тестов $\{b/B\}$. С учетом принятых допущений любое отображение модели знаний k можно представить следующим образом [3]:

$$f/\mu_f: \{b/B\} \rightarrow a/A, \text{ где } \{b/B\} \in W_f(\{b/B\}), \mu_f \in \{\mu\}_f \quad (2)$$

или в альтернативной записи:

$$a/A \downarrow_f \{b/B\}, \text{ где } \{b/B\} \in W_f(\{b/B\}). \quad (3)$$

Разные механизмы могут реализовывать, например, разные агенты [6]. Запись $a/A \downarrow \{b/B\}$ на $W(\{b/B\})$ будем понимать как существование механизма в рамках фиксированного исчисления, позволяющего получить нужный результат.

Базовым доменом является домен, с помощью которого описываются максимально точные значения теста (область возможных значений - ОВЗ). Для таких тестов как «Температура», «Длина», «Масса», «Скорость», «Ускорение», «Сила», «Время» в качестве базового домена можно принять отрезок числовой прямой. Для теста «Возраст» базовым доменом будет целочисленный интервал от 0 до 120 и т.д. Доменная структура всех перечисленных выше тестов является фракталом.

Дискретный домен максимальной общности назовем N -арным конструктом теста, где N – число элементов домена. Примеры бинарных конструктов: $\{+, -\}$, $\{\text{Норма: Отклонение}\}$, $\{\text{Эффективно; Не эффективно}\}$, $\{\text{Жалобы есть; Жалоб нет}\}$, $\{\text{Трудоспособный возраст; Не трудоспособный возраст}\}$ и т.д. Для одного и того же теста может быть задано сколь угодно много N -арных конструктов. Между базовым доменом теста и N -арным конструктом также может быть задано сколь угодно много промежуточных доменов. Внутри иерархии доменов задаются правила пересчета элементов одного домена в элементы домена более высокого уровня общности. Правила пересчета могут быть самыми разными: детерминированными, нечеткими, нейросетевыми и т.д.

Конкретную иерархию доменов (фрактал), основанием которой служит базовый домен, вершиной - N -арный конструкт, с фиксированными правилами пересчета назовем *фрактальным конфигуратором*. Фрактальный конфигуратор теста τ будем обозначать следующим образом:

$$fr_\tau: OBZ(\tau) \succ \delta_N(\tau), \quad (4)$$

где $OB3(\tau)$ - базовый домен, $\delta_N(\tau)$ - N -арный конструкт. Множество конфигураторов для теста τ обозначим $\{fr\}_\tau$. Аналогичным образом можно задать фрактальный конфигуратор для произвольного критерия K с бесконечным множеством значений на базовом уровне, который обозначим fr_K . Конфигуратор отражает характерную для синергетики ситуацию «расширение - сжатие». Конфигураторы могут быть стохастическими и/или динамическими, т.е. меняться со временем. На основе фрактальных конфигураторов строятся ориентированные графы доменов в методе предельных обобщений [7]. Детерминированные фрактальные конфигураторы тестов и критериев могут быть описаны в многоцелевом банке знаний [5].

Пример конфигулятора для теста «Температура тела» (D1 – D4 – домены, метки задают правила пересчета из одного домена в другой):

Температура тела {

D1 { Нормальная температура тела ^N;
Ненормальная температура тела ^b c}
D2 { Пониженная температура тела ^b;
Нормальная температура тела ^N;
Повышенная температура тела ^c}
D3 { Резкая гипотермия ^b [34.0; 34.9];
Гипотермия ^b [35.0; 35.9];
Нормальная температура тела ^N [36.0; 36.9];
Субфербильная температура ^c [37.0; 37.9];
Фербильная температура ^c [38.0; 39.9];
Предельная температура ^c [40.0; 42.0]}
D4 { [34.0; 42.0]}}

Примеры других конфигураторов:

Жалобы на боли в области живота {

D1 {Болей в животе нет (норма) ^N; Жалобы на боли в животе ^b}
D2 {Болей в животе нет (норма) ^N; Постоянные ноющие боли в правом подреберье ^b;
Схваткообразные боли в правом подреберье ^b; Опооясывающие боли в эпигастрии ^b;
Боли в левом подреберье ^b}}

Аппетит {

D1 {Аппетит нормален ^N; Аппетит изменен ^b}
D2 {Отвращение к пище (анорексия) ^b; Аппетит снижен ^b; Аппетит нормален ^N;
Аппетит повышен ^b}}

Конфигураторы тестов представляют собой лексические деревья, как и большинство объектов Многоцелевого банка знаний [5]. Конфигураторы хранятся в текстовых файлах.

Таким образом, практически любой тест и критерий является мультифракталом (задает множество конфигураторов). Данный вывод позволяет по иному взглянуть как на интерфейс приложения, так и на организацию хранения и обработки данных. Без потери общности любую форму ввода данных можно определить следующим образом:

$$form ::= \{ \tau / fr_\tau \}. \quad (5)$$

Данная нотация означает, что для любого теста из формы необходимо указывать конкретный конфигуратор, с помощью которого выбирается уровень общности и значение теста. Пример формы:

Дневник ::= {Время/Т; Температура/ТЕМП; Пульс/ПУЛЬС;...}, где Т, ТЕМП, ПУЛЬС – фрактальные конфигураторы.

Для демонстрации работы с формами вида (5) разработана специальная программа.

Программа позволяет выбрать нужную форму из списка (конфигураторы форм хранятся в текстовых файлах), выбрать домен для каждого теста и выбрать значение из домена. Результаты сохраняются в XML-файле. Важно отметить, что при сохранении данных для каждого теста автоматически вычисляются все значения в доменах более высокого уровня общности, чем домен заданный пользователем. Например, если температура тела задана в градусах (домен D4), то при сохранении автоматически вычисляются значения в доменах D3, D2 и D1. Подобный пересчет позволяет применить отображения вида (2).

Модель знаний (1) вида (2) также хранится в текстовом файле. Формат произвольного отображения с использованием Банка тестов (ФО – функциональное отображение) [4, 5]:

```
<Полное имя ФО> -  $f/\mu_f$  {ФО_<Аббревиатура для ФО>
    Вход {inp_
        <входные тесты> - { $b/B$ }}
    Выход {out_
        <выходные тесты> -  $a/A$  }
    Методы {met_
        <методы обработки> - { $\mu$ }_f}}
Семантическая группа отображений:
<Имя предметной области> {ПРО_<СемантИмя>
    [Определения {def_
        <Сокращенное имя теста> - <Полное имя теста> [{Ссылка}]
        ....
    }]
    <ФО_1>
    ...
    <ФО_N>
} ПРО
```

Заключение. Представленные инструменты позволяют в фазе расширения выявить и задать параметры порядка $\{p/P\}$ – результат интеллектуального синергетического управления. Параметром порядка максимального уровня общности является модель поведения динамической системы [3]. Модель поведения определяется в результате решения задачи диагностики [8]. Пример – модель пациента [3, 4].

Литература

- Осипов Г.С. Динамические интеллектуальные системы // Искусственный интеллект и принятие решений. – М. : URSS, 2008, № 1. – С.47 - 54.
- Колесников А.А. Синергетические методы управления сложными системами: теория системного синтеза. - М.: Едиториал УРСС, 2006. - 240 с.
- Прокопчук Ю.А. Интеллектуальные медицинские системы: формально-логический уровень. – Дн-ск: ИТМ НАНУ, 2007.- 259 с.
- Алпатов А.П., Прокопчук Ю.А., Юденко О.В., Хорошилов С.В. Информационные технологии в образовании и здравоохранении. – Днепропетровск : ИТМ НАНУ и НКАУ, 2008.- 287 с.
- Прокопчук Ю.А. Многоцелевой банк знаний в области клинической медицины // Вестник ХНТУ, 2008. - №1(30). – С. 14 – 20.
- Прокопчук Ю.А. Проведение врачебных консилиумов с участием интеллектуальных систем // Вестник ХНТУ, 2007. - №4(27). – С. 198 – 202.
- Прокопчук Ю.А. Метод предельных обобщений – эффективный принцип работы вычислительного интеллекта // Искусственный интеллект, 2008. - №3. - С. 727 – 736
- Прокопчук Ю.А., Белецкий А.С. Формальные схемы сценариев диагностики проблемных состояний сложных систем // Збірник доповідей науково-практичної конф. «Системи підтримки прийняття рішень. Теорія і практика». – Київ: ІПММС НАНУ, 2008. – С. 21- 24.