ПРИМЕНЕНИЕ АЛГЕБРАИЧЕСКОГО ПОДХОДА

В КВАЛИМЕТРИИ МОДЕЛЕЙ И ПОЛИМОДЕЛЬНЫХ КОМПЛЕКСОВ

Микони С.В., Соколов Б.В.. Юсупов Р.М. (Санкт-Петербург)

Введение

Анализ многочисленных техногенных аварий и катастроф, происходящих в современном мире, показывает, что одна из главных причин возникновения перечисленных явлений связана с усилением роли факторов сложности в существующих и проектируемых организационно-технических системах, используемых в различных предметных областях. При этом, говоря о проблемах сложности современных объектов-оригиналов (реальных и абстрактных), принято выделять следующие основные аспекты сложности: *структурную сложность, сложность функционирования, сложность принятия решений и выбора сценариев поведения, сложность развития, сложность их формального описания и моделирования* [1]. К настоящему времени наука создала богатый методологический и методический аппарат, позволяющий успешно преодолевать трудности, связанные с воздействием факторов сложности в современном мире. В основу этого аппарата положена системная отрасль научных знаний [1], одной из компонент которой является квалиметрия моделей и полимодельных комплексов, представляющая из себя прикладную науку, ориентированную на разработку методологии и технологий оценивания качества указанных моделей и комплексов [2,4,6,7]. Следует отметить, что в области оценивания качества моделей к настоящему времени получено много интересных научных и практических результатов, связанных как с количественным и качественным оцениванием и анализом таких свойств моделей как адекватность, сложность, неопределенность [2-3,5-7], так и с упорядочением и выбором (синтезом) моделей для решения заданных классов задач [5-8].

При этом для различных предметных областей создавались свои теории и технологии моделирования, разрабатывались и разрабатываются огромное количество банков моделей и полимодельных комплексов, которые широко используются на практике [5-7]. Вместе с тем, при наличии большого разнообразия моделей остаются открытыми вопросы обоснованного выбора моделей, сравнения различных технологий моделирования [5-7]. При решении перечисленных задач важную роль играют, во-первых, задачи выбора универсальных формальных средств, позволяющих на едином метаязыке описывать различные классы и виды моделей и полимодельных комплексов, и, во-вторых, задачи обоснования и выбора системы показателей качества и соответствующих методик многокритериального оценивания, анализа и упорядочения исследуемых моделей и полимодельных комплексов.

Ранее указывалось, что весьма перспективным в этом отношении является подход к описанию моделей, базирующийся на понятии математической структуры [1,6,7]. В докладе проведена конкретизация данного подхода на основе использования многосортной алгебраической системы в языке предикатов первого порядка. Понятие сортности алгебраической системы использовано для оценивания таких свойств моделей как их структурная и функциональная сложность, а также при оценивании свойства достоверности результата моделирования, конкретизируемого по отношению к классам решаемых задач.

1. Модель объекта, метода, задачи

В качестве математической основы описания рассматриваемых в докладе объектов-оригиналов (в том числе и самих моделей, описывающих указанные объекты, которые могут иметь реальную и виртуальную природу) выберем алгебраическую систему (структуру). Она представляет собой множество A (носитель) с заданным на нём набором операций и отношений (сигнатурой) и удовлетворяющим некоторой системе аксиом [9]. Алгебраическая система представляется следующей четвёркой символов:

$$<A, C, P, F>. \qquad (1)$$

Эти символы представляет собой множества: A – предметных переменных (носитель); C – констант; F – функций; P – предикатов.

Многосортная алгебраическая система является моделью языка первого порядка Ω [9]. Её элементы – суть результаты интерпретации этого языка с применением функций D, $Cnst$, Fn, Pr: D: $\pi \to A_\pi$, $\pi \in Srt$; $Cnst$: $cnst \to c$; Fn: $fn \to f$; Pr: $pr \to p$.

Элемент π множества Srt называется *сортом* объекта. Ему соответствует *тип* переменной в программировании. Для каждого сорта фиксируется множество предметных переменных $a^\pi_1,\dots,a^\pi_n \in A_\pi$ (носитель[1]) и констант $c^\pi_1,\dots,c^\pi_k \in C_\pi$. Множество констант C_π характеризует *диапазон значений* соответствующей переменной (домен в базе данных). Множества функций F и предикатов P образуют сигнатуру модели. Каждый предикат p из множества предикатов P может быть выражен через отношение r. Поэтому в другой записи четвёрка символов (1) представляется через отношения: $<A, C, R, F>$.

Алгебраическая система членится (по А.И. Мальцеву) на модель (реляционную систему) $<A, C, R>$ и алгебру $<A, C, F>$. Модель $<A, C, R>$ представляет собой совокупность отношений на носителе A, отражая структуру *объекта-оригинала*. Любая операция $f \in F$ алгебры $<A, C,F>$ реализуется с помощью некоторого алгоритма, являющегося моделью *метода* вычислений. Таким образом, модель должна отражать как структуру объекта-оригинала, так и реализуемый ею метод вычислений.

Любое математическое выражение с переменными следует рассматривать как модель-*прототип*, имеющую множество решений – моделей-*экземпляров* в терминологии искусственного интеллекта. Эти решения могут быть получены *разными* методами, в то же время применимыми к разным моделям-прототипам.

Множественность задач на одной модели-прототипе можно проиллюстрировать на такой простой закономерности, в силу всеобщности называемой законом, как закон Ома. Любая из трёх связанных им переменных может быть получена через две другие. А это значит, что на основе трёх переменных можно сформулировать три задачи. Иными словами, существуют три модификации формулы. Одна из переменных вычисляется с помощью операции умножения, а две остальных – с помощью операции деления.

Таким образом, моделью, объединяющей модель-прототип и модель-экземпляр, является ***модель задачи***. Их соотношение можно выразить следующей формулой:

модель задачи = модель-прототип + модель метода → модель-экземпляр.

Известно, что модель и метод решения задачи *взаимозависимы*, что наглядно демонстрируется на логико-комбинаторных задачах. Простейшим примером такой взаимозависимости является определение делимости целого числа на 4. При десятичном представлении числа необходимо выполнять деление n-разрядного десятичного числа поразрядно n_{10} раз. Сложность этого процесса пропорциональна числу разрядов $O(n_{10})$. При двоичном представлении числа достаточно проанализировать два младших разряда на 0. Сложность этого процесса равна двум $O(2)$. Налицо уменьшение сложности на n_{10}–2.

Однако условием уменьшения сложности является необходимость преобразования числа из удобной для человека десятичной системы в удобную для компьютера двоичную систему, что может рассматриваться как ***перевод*** с одного языка представления объекта (числа) в другой. С точки зрения компьютерной науки это preprocessing. Число шагов

[1] Носитель можно считать вырожденной алгебраической системой с пустым набором операций и отношений.

перевода из десятичного числа с n_{10} разрядами в двоичное число с n_2 разрядами равно: $n_2=\log_2 10\cdot n_{10}=3{,}32\cdot n_{10}$. Эта величина и определяет сложность преобразования, т.е. перевода с языка десятичной системы счисления на язык двоичной системы счисления. В качественном смысле имеет место закон сохранения сложности – чем менее сложен метод решения, тем более сложен процесс получения подходящей для этого метода модели. Количественно *полный* перевод десятичного числа в двоичное число, исходя из соотношения выражений n_{10}–2 и $3{,}32\cdot n_{10}$, невыгоден. Это соответствует незнанию того, что для решения задачи достаточно перевести только младший разряд десятичного числа в 2 разряда двоичного числа, которые и нужны для решения поставленной задачи.

Рассмотрение этого простейшего примера показывает, что решение поставленной задачи «в лоб», без предварительного исследования её особенностей, эквивалентно полному перебору. Выявление закономерности, присущей решаемой задаче, позволяет перейти от полного перебора к сокращённому перебору. Здесь важно учесть фактор времени. При решении задачи «в лоб» затрачивается минимальное время для создания модели, а на выявление закономерности, способствующей ускорению решения задачи, требуется дополнительное время, которое может быть оправдано массовым решением задачи, либо её уникальностью.

Из рассмотренного примера следует, что время моделирования включает в себя время, затрачиваемое на *проектирование модели,* и *время решения* задачи выбранным методом. Эти показатели противоречивы. Чем больше времени уходит на проектирование модели с целью выявления закономерностей, тем меньше времени может занять решение задачи.

Другим примером упрощения метода решения задачи является метод Монте-Карло. Фактически он никак не отражает особенности модели в силу своей универсальности. На его реализацию требуется минимум времени. Но зато процесс решения характеризуется *экспоненциальной* сложностью и, в силу этого, для сложных систем не удаётся получить приемлемую точность результата решения задачи за приемлемое время. Эта проблема в разной степени свойственна всем стохастическим методам. Поэтому они, как правило, применяются для приближённого решения задач.

Повышения точности решения можно добиться за счёт более детального изучения свойств объекта, что во многих случаях достигается с применением детерминированного подхода. Закономерности, найденные в процессе изучения свойств объекта, позволяют также сократить время решения задачи за счёт сокращённого перебора, применяя алгоритмы *полиномиальной* сложности.

Известно, что метод решения задачи может быть реализован с применением различных алгоритмов. Их качество оценивается вычислительной сложностью, под которой понимается максимальное число шагов, необходимых для решения задачи.

Из изложенного следуют два пути увеличения точности решения задачи: усложнение модели на основе учёта выявленных закономерностей, либо усложнение метода (алгоритма) решения задачи. Выбирается тот путь, который удовлетворит требования к точности результатов и времени моделирования. В самом общем виде методы решения задач можно разделить на следующие группы (табл. 1).

Общие методы решения задач Таблица 1

Наименование метода	Трудоёмкость	Достоверность	Универсальность
Псевдослучайная последовательность	Определяется объёмом выборки	Пропорциональна объёму выборки	Универсален
Полный перебор	Экспоненциальная	Полная	Универсален
Сокращённый перебор	Полиномиальная	Полная	Специализирован

2. Оценивание сложности моделей и полимодельных комплексов

2.1 *Сложность однородных моделей*

На основании алгебраической системы, как модели языка первого порядка, можно оценить сложность конкретных математических моделей безотносительно их специфики.

Представим множество констант C как домен значений переменной $|D|=N$. Параметр N характеризует *информационную* сложность переменной. В порядке возрастания информационной сложности он делится на 4 категории: два значения (2), конечное число значений (k), счётное (k_∞) и бесконечное множество значений (∞).

Представим носитель A конечным множеством X мощности $|X|=m$. Тогда прямое (декартово) произведение $X\times X\times\ldots\times X = X^n$ упорядоченных наборов или n-ок $(x_1, x_2,\ldots,x_n)$ задаёт n–арный универсум. При n=2 универсум $X\times X$ представляется полным графом с петлями.

В n-местное отношение $R^n\subseteq X^n$ входят часть n-ок (в частном случае все n-ки) из n-местного декартового произведения (универсума) X^n. Важным частным случаем является бинарное отношение $R^2=R$, состоящее из p упорядоченных пар вида (x_i, x_j). Оно представляется графом G, число связей p в котором и характеризует *структурную* сложность модели. Нуль-граф $R=\varnothing$, в котором отсутствуют упорядоченные пары (дуги) между вершинами, имеет нулевую сложность p=0. Полный граф, содержащий все связи между вершинами («каждая соединена с каждой и с собой»), имеет максимальную структурную сложность 2-го порядка $p=m\times m=m^2$. Структурная сложность n-го порядка (n-местное отношение R^n) интерпретируется множеством путей длины n, соединяющих вершины графа. Таким образом, структурная сложность модели может быть охарактеризована вектором $\mathbf{c}_{ctr}=(m,\ p,\ p^n)$. Если модель содержит l отношений, то структурная сложность характеризуется матрицей размерностью $l\times 3$.

Пусть домен D значений переменной представляет области определения и значений функции f: $D\times\ldots\times D\rightarrow D$. Тогда *функциональная* сложность модели определяется видом функции, её арностью (числом аргументов) n и числом функций k в модели. Сложность *алгебраической* функции определяется максимальной степенью её переменных $s_{\max}$.

Сложность *неэлементарной* функции определяется максимальной степенью $s_{\max}$ члена ряда, в который разлагается эта функция в области сходимости. Это число определяет и точность аппроксимации функции.

Поскольку компьютерная обработка выполняется в численном режиме, число $s_{\max}$ всегда *конечно*. Следует только помнить, что число $s_{\max}$ зависит от численного метода реализации функции и не может быть определено по её виду. Это лишний раз показывает, что для компьютерного моделирования недостаточно представления модели в аналитической форме записи. Необходимо её рассматривать вместе с методом реализации модели. Тогда, имея численные характеристики отношений и функций, существует возможность численно оценить сложность модели в совокупности с реализующим её методом. Раздельно сложность модели и метода оцениваются соответственно по объему используемой памяти компьютера и числу шагов алгоритма (вычислительная сложность).

Тогда функциональная сложность модели характеризуется вектором $\mathbf{c}_f=(s_{\max},\ n)$, а сложность k-функциональной модели – матрицей размерностью $k\times 2$.

Общая сложность модели характеризуется вектором $\mathbf{c}=(N,\ m,\ p,\ p^n,\ s_{\max},\ n)$. Это позволяет по векторным оценкам сложности моделей находить множество недоминируемых методов (множество Парето). Однако при малом числе сопоставляемых методов и предложенных шести признаках велика вероятность того, что все они войдут в множество Парето.

Для получения линейного порядка на множестве моделей требуется синтезировать обобщённую оценку размерности модели как количественную характеристику её сложности. Однако при переводе векторной оценки модели в скалярную оценку возникает проблема оценивания важности показателей размерности. Она может быть решена только экспертным способом.

Здесь могут быть учтены следующие соображения. Каждая переменная отражает *одно* из элементарных *свойств* объекта-оригинала. Число значений характеризует *точность* представления этого свойства. Бинарное отношение характеризует *структуру* объекта, а *n*-арное отношение – *взаимодействие* его элементов. Функция одного аргумента отражает *элементарную закономерность*, выявленную в объекте-оригинале, а *n*-арная функция отражает *n элементарных закономерностей*.

С точки зрения представления знания переменные соответствуют базовым понятиям предметной области, а отношения и функции отражают присущие ей структуры и процессы. При таком истолковании языка модели можно экспертным способом найти количественные оценки важности показателей её размерности, например, с помощью аппарата матриц парных сравнений. Остаётся открытым вопрос о зависимости важности показателей размерности модели относительно решаемой задачи. Можно предположить, что этой зависимостью во многих случаях можно пренебречь. Тогда предлагаемый способ оценивания сложности можно считать независимым от решаемой задачи. Полученная оценка сложности может быть включена в комплексную оценку качества модели и подлежать минимизации при его оценивании.

2.2 *Сложность полимодельного комплекса*

При определении сложности полимодельного комплекса, в который входят модели, представленные на языке одного сорта, следует учитывать их взаимодействие. В рамках языков одного сорта (уровня) оценивается сложность операции *подстановки*.

При оценивании полимодельного комплекса, включающего по терминологии многосортной алгебраической системы модели *разного* сорта, необходимо учитывать помимо сложности самих моделей сложность перевода с языка одного сорта на другой. Для оценивания его сложности будем использовать модель с иерархической структурой сортов моделей.

При переходе от более простого языка к более сложному языку применяется операция *группировки*, а при обратном переходе – операция *свёртки*. Поясним это на следующих примерах. При переходе от скалярных оценок к векторной оценке используется операция группировки (конкатенации). При обратном переходе вектор свёртывается в скаляр с помощью синтезирующей функции. При переходе от векторных оценок к матричной оценке выполняется объединение векторов в матрицу. При умножении матрицы на вектор получаем вектор.

Таким образом, при оценивании двухсортной модели к оценкам сложности модели каждого сорта должны добавляться функции перевода с языка одного сорта модели на другой, как прямые, так и обратные при двустороннем взаимодействии. Этой идее отвечает формула:

Сложность двухсортной модели = сложность модели сорта 1 + сложность модели сорта 2 + сложность перевода (1, 2) + сложность перевода (2, 1).

Аналогичным образом, оценивается сложность модели с большим числом сортов, как полимодельного комплекса. При этом очень важно проверять каждый раз корректность перехода от модели к модели. В докладе для важной прикладной задачи, связанной с управлением активными подвижными объектами (АПО), предложен пример функториального перехода из категории орграфов *Kat* ***Ф***, задающей структурные (статические) модели технологий выполнения работ (операция) с АПО, в категорию динамических моделей $Kat\,D$, описывающих собственно процессы выполнения взаимосвязанных операций, связанных с функционирование системы АПО. В этом случае конструирующий ковариантный функтор $G: \Phi \to D$ устанавливает соответствие как между вершинами графа $x_i \in Ob\,\Phi$ и динамическими моделями $d_i \in Ob\,D$, так и между рёбрами графа $<x_i, x_j> \in Mor_\Phi(X,X)$ и отображениями динамических моделей $\psi_{ij} \in Mor_D(G(<x_i, x_j>))$, названными в [7] отображениями сопряжённости. В рамках предложенной формализации проверка условий функториальности преобразования орграфа в динамическую модель может, например, базироваться на анализе результатов транзитивного замыкания бинарных отношений, задающих в каждой из моделей очерёдность следования операций в СУ АПО. Для графовой (статической) модели проверка данных условий не вызывает особых затруднений и сводится к выполнению простейших действий с матрицей смежности. Для динамической модели данная проверка требует проведения специальных преобразований её структуры [6-7]. Проведённые исследования показали, что в рамках предложенного варианта полимодельного описания функционирования СУ АПО, выполняются не только условия функториальности, но и условия общности положения отображений сопряжённости [6-7].

3. Показатели оценивания качества модели

Рассмотрим показатели, которые численно характеризуют свойства модели. Основным свойством модели-прототипа является *адекватность* (соответствие) объекту-оригиналу [3]. Сама по себе математическая модель не зависит от области применения и в этом смысле является универсальной. Поэтому её соответствие оригиналу следует рассматривать по отношению к поставленной задаче. Относительно конкретной задачи модель может быть неполной, избыточной и полностью соответствующей поставленной задаче. Очевидно, что модель, избыточная по отношению к некоторой задаче, может полностью соответствовать более сложной задаче. Если принять модель максимальной сложности за целое, то её количественный эквивалент может использоваться для нормирования оценок более простых моделей.

Модель метода принято оценивать вычислительной сложностью, которая характеризует трудоёмкость моделирования. Показателем трудоёмкости получения результата является асимптотическая сложность алгоритма, определяющая рост числа его шагов с ростом размерности задачи.

Модель-экземпляр следует оценивать по *достоверности*. Она определяется числом знаков после запятой. Достоверность результата конкретизируется относительно типа решаемой задачи: точность для численных задач; отношение к глобальному оптимуму для оптимизационных задач; вероятность при решении статистических задач; коэффициент уверенности при решении логических задач.

Этой систематизации отвечает таблица 2, составленная для трёх моделей.

Модель задачи — Таблица 2

Модель объекта	Адекватность			Размерность модели		Модель метода	Трудоёмкость	Достоверность
	Св-во 1	……..	Св-во N	n	m		$O(n, m)$	
МО1						ММ1		
МО2						ММ2		
МО3						ММ3		

Здесь n – число вершин графа, а m – число связей (дуг, рёбер).

Наиболее сложно определить отношение локального оптимума к глобальному. Для этого *надо знать* глобальный оптимум. Точность решения статистических задач оценивается доверительной вероятностью. Коэффициент уверенности в логических задачах принадлежит интервалу [0, 1]. Коэффициент уверенности k логического вывода в двоичной системе равен k=1, а в многозначном логическом выводе $k \le 1$. Но булевы логические операции неприменимы в условиях неопределённости. Иными словами, многозначный логический вывод имеет большую область применения, чем двоичный, поскольку сводится к двоичному выводу в отсутствие неопределённости. Для полимодельного комплекса число таблиц определяется числом моделей, входящих в комплекс. К ним добавляется таблица преобразования данных между взаимодействующими моделями.

4. Многокритериальное оценивание качества модели

Наиболее универсальным методом многокритериального оценивания является метод отклонения от цели [10]. Цель описывается вектором численных оценок требований. Каждая компонента вектора представляет собой частную цель.

Представим цель в n-мерном пространстве признаков вектором $\mathbf{c}=(c_1,\ldots,c_j,\ldots,c_n)$. Назовём её *образцом*. Введём меру обобщённого отклонения от цели, которая позволяет не только найти вариант, ближайший к образцу, но и упорядочить объекты по удалённости от него. Для простоты вначале рассмотрим образец, чьи свойства формируются ограничениями по равенству ($y_j=c_j$). Очевидно, что отклонение j-го признака объекта x_i в любую сторону от точки c_j ($c_j\pm\Delta y_j$) определяет меру удалённости по этому признаку объекта от цели.

Поскольку знак величины отклонения объекта $x_i\in X$ по j-му признаку от частной цели c_j не имеет значения, определим относительное отклонение j-го признака от цели как:

$$\delta y_{ij}^{p}=\begin{cases}\dfrac{\left|y_{ij}-c_j\right|}{y_{j,\max}-c_j}; & y_{ij}>c_j\\[2ex] \dfrac{\left|y_{ij}-c_j\right|}{c_j-y_{j,\min}}; & y_{ij}<c_j\end{cases} \qquad (2)$$

При использовании интервального ограничения $y_j\in[c_{\text{н},j},\ c_{\text{в},j}]$ относительное отклонение равно 0 ($\delta y^{\text{и}}_{ij}=0$), если все точки интервала равноценны. При выходе значения y_{ij} за любую границу интервала $[c_{н,j},\ c_{в,j}]$ относительное отклонение определяется по формуле (2), где вместо c_j подставляется ближайшая граница интервала $[c_{н,j},\ c_{в,j}]$.

В отличие от точечного и интервального критериев при использовании полуинтервальных ограничительных критериев $y_j\geq c_j$ и $y_j\leq c_j$ отклонение от порогового значения c_j может трактоваться двояко: и как мера нарушения требования c_j, и как мера его превышения. На рис. 1 зона нарушения для ограничения снизу ($y_j\geq c_j$) находится слева от требования c_j, а зона превышения – справа от него.

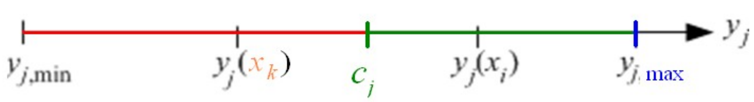

Рис.1. Зоны нарушения и превышения требования c_j для ограничения снизу ($y_j\geq c_j$)

В зоне *нарушения* $[y_{j,\min},\ c_j]$ невыполнение требования c_j оценивается разностью $c_j - y_j(x_k)$, которой ставится в соответствие штрафная санкция. Напротив, если значение признака $y_j(x_i)$ превосходит ожидаемое значение c_j, разность $c_j - y_j(x_i)$ с противоположным знаком должна отражать величину *поощрения.* Для ограничения сверху ($y_j\leq c_j$) зоны нарушения и поощрения меняются местами.

Нормируя отклонение от требования c_j соответствующими отрезками шкалы, получаем формулы относительного отклонения для ограничения снизу:

$$\delta y_{ij}^{\text{н}} = \begin{cases} \dfrac{c_j - y_{ij}}{y_{j,\max} - c_j}; & y_{ij} > c_j \\ \dfrac{c_j - y_{ij}}{c_j - y_{j,\min}}; & y_{ij} < c_j \end{cases} \tag{3}$$

и формулы относительного отклонения для ограничения сверху:

$$\delta y_{ij}^{\text{в}} = \begin{cases} \dfrac{y_{ij} - c_j}{y_{j,\max} - c_j}; & y_{ij} > c_j \\ \dfrac{y_{ij} - c_j}{c_j - y_{j,\min}}; & y_{ij} < c_j \end{cases} \tag{4}$$

При попадании значения y_{ij} в зону нарушения относительное отклонение δy_{ij} оказывается положительным, а при попадании в зону превышения – отрицательным (см. рис.2).

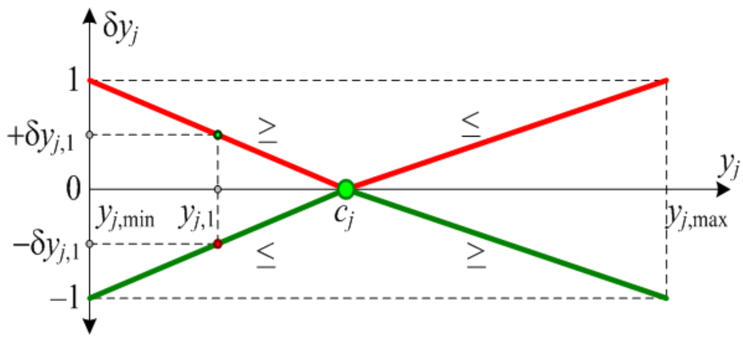

Рис.2. Шкала отклонений для ограничений «снизу» и «сверху»

Так, например, значение j-го признака $y_{j,1}$ при ограничении снизу попадает в зону штрафов ($+\delta y_{j,1}$), а при ограничении сверху – в зону поощрений ($-\delta y_{j,1}$).

Таким образом, относительные отклонения $\delta y^{\text{н}}_{ij}$ и $\delta y^{\text{в}}_{ij}$ измеряются в полярной шкале [–1, +1], где значения относительных отклонений в шкале [0, +1] рассматриваются как штрафы, а в шкале [–1, 0] – как поощрения.

При подборе функции, синтезирующей отклонения по всем ограничительным критериям, следует учитывать знаки отклонений, отражающие штрафные санкции и поощрения. Этому условию отвечает аддитивная обобщающая функция, представляющая собой алгебраическую сумму отклонений по ограничительным критериям всех типов:

$$\delta y_i = \sum_{j=1}^{n_{\text{н}}} w_j \, \delta y_{ij}^{\text{н}} + \sum_{j=1}^{n_{\text{в}}} w_j \, \delta y_{ij}^{\text{в}} + \sum_{j=1}^{n_{\text{и}}} w_j \, \delta y_{ij}^{\text{и}} + \sum_{j=1}^{n_{\text{р}}} w_j \, \delta y_{ij}^{\text{р}}$$

(5)

Верхние индексы сумм обозначают количество ограничительных критериев соответствующего типа, причём $n_{\text{н}}+n_{\text{в}}+n_{\text{и}}+n_{\text{р}}=n$, где n – общее число критериев, а w_j – важность (вес) j-го признака, $\sum_{j=1}^{n} w_j = 1$. Если все критерии имеют равную важность, их веса равны $w_j=1/n$, $j=\overline{1,n}$.

Величина δy_i представляет собой *алгебраическую* сумму относительных отклонений. Значение функции (5) складывается из положительной части (сумма штрафов) и отрицательной (сумма отклонений). Функция положительна, если сумма штрафов превышает сумму поощрений и отрицательна – в противном случае. Функция δy_i достигает граничного значения –1, если *все* отклонения от локальных целей отрицательны и граничного значения +1, если *все* относительные отклонения положительны. Функция принимает значение 0 либо в случае точного достижения всех локальных целей, либо когда ненулевая сумма относительных штрафов равна сумме относительных поощрений (эффект компенсации).

Учитывая поощрительный характер отрицательных отклонений, объекты упорядочиваются в направлении от максимального отрицательного отклонения $\delta y_{\min}<0$ к максимальному положительному отклонению $+\delta y_{\max}$. Поскольку отрицательная величина δy_i не отвечает аксиомам метрики, в случае $\delta y_{\min}<0$ все оценки можно свести к положительным, пересчитывая их по формуле:

$\delta y_i' = \delta y_i - \delta y_{\min}$.

Наилучшим считается объект x^*, обладающий минимальным значением обобщённого отклонения от цели:

$$x^* = \arg(\min_{x \in X}(\delta y_i))$$

(6)

Остальные объекты упорядочиваются в направлении возрастания функции отклонения от обобщённой цели δy_i, $i = \overline{1, N}$. Отбору подлежит объект x^* и, если требуется, заданное количество следующих за ним объектов.

В тех случаях, когда все ограничительные критерии относятся только к интервальному или точечному типу, чему отвечают 2 последних слагаемых формулы (5), либо нас интересуют только штрафы или только поощрения, знак отклонения может быть только положительным. Это позволяет использовать в качестве обобщающих функций как аддитивные, так и мультипликативные обобщающие функции.

Метод отклонения от цели позволяет оценивать объекты, не удовлетворяющие предъявленным требованиям, и ранжировать их по степени приближения к этим требованиям. В частном случае, когда компоненты вектора целей совпадают с границами шкал показателей, метод отклонения от цели сводится к обычной скалярной оптимизации.

5. Пример оценивания качества модели

В качестве примера рассмотрим способы решения задачи нахождения расстояния от *i*-й вершины графа до всех остальных вершин.

I. **Модель объекта** – матрица расстояний *D*.

Пример матрицы расстояний *D*:

$$\mathbf{D}_1 = \begin{pmatrix} 0 & 9 & 6 & \infty & 3 \\ 9 & 0 & \infty & \infty & \infty \\ 6 & \infty & 0 & \infty & \infty \\ \infty & 2 & 4 & 0 & \infty \\ \infty & 4 & \infty & \infty & 0 \end{pmatrix}$$

II. **Методы** (алгоритмы) **решения**

1. Умножение *i*-й строки графа расстояний на матрицу расстояний по формуле:

$$d_{ik}^s = \min\left\{ d_{ik}^{s-1}, \min_j\left(d_{ij}^{s-1} + d_{jk}^1\right)\right\},\ i, j = \overline{1, n}. \qquad (7)$$

2. Алгоритм динамического программирования Дейкстра

с последовательным уменьшением расстояния между вершинами v_i и v_k по формуле:

$$d_{ik}^s = \min\left\{d_{ik}^{s-1}, \min_j\ d_{ij}^{s-1} + d_{tk}\right\};\ t = \arg\left(\min_j\ d_{ij}^{s-1}\right). \qquad (8)$$

3. Нахождение всех вариантов путей от i-й вершины до всех остальных на основе модифицированной матрицы смежности с последующим вычислением минимальных расстояний [11-12].

III. Расчёт асимптотической сложности методов решения

1. ***Умножение i-й строки на матрицу по формуле*** (7)

Число операций: Умножение строки на столбец – n; нахождение минимальной суммы – n–1; сравнение с предыдущим расстоянием – 1. Общее число операций умножения строки на столбец равно: $n+n-1+1=2\cdot n$. Умножение строки на n столбцов: $2\cdot n\cdot n=2\cdot n^2$; Число умножений i-й строки на матрицу до достижения всех вершин – n–1; Общее число операций равно $\mathbf{2\cdot n^2\cdot(n-1)}$.

2. ***Алгоритм динамического программирования Дейкстра***

Число операций: Нахождение вершины с минимальным расстоянием от предыдущей – n–1; добавление расстояния между этими вершинами к общему расстоянию от начальной вершины – 1; исключение помеченной вершины из рассмотрения – 1. Общее число операций нахождения ближайшей вершины равно: n+1; За счёт исключения помеченной вершины общее число операций при поиске каждой следующей вершины равно $n+1-i$;

$i=\overline{1,n-1}$. Общее число операций за n итераций равно: $\sum_{i=1}^{n-1} n+1-i$ $=\mathbf{n\cdot(n-1)}$.

3. ***Нахождение расстояний по всем путям от i-й вершины до всех остальных на основе модифицированной матрицы смежности***

Число операций: умножение строки на столбец – n; выявление и исключение циклических маршрутов – 1. Общее число операций умножения строки на столбец: n+ 1. Умножение строки на n столбцов: $(n+1)\cdot n$. Число умножений i-й строки на матрицу до достижения всех вершин – n–1. Общее число операций на 1-м этапе равно $(n+1)\cdot n\cdot(n-1)=n\cdot(n^2-1)$. Вычисление расстояний для путей любой длины – $n\cdot(n-1)/2$; Общее число операций на двух этапах равно $\mathbf{n\cdot(n^2-1)+n\cdot(n-1)/2}$.

Первые два метода имеют общую модель объекта и ориентированы только на вычисление расстояний. Поэтому они сравниваются только по трудоёмкости. В отличие от нахождения расстояний перемножением строки на матрицу, реализующего процедуру параллельного поиска, алгоритм динамического программирования реализует метод *сокращённого* последовательного поиска, в силу чего его трудоёмкость в $\mathbf{2\cdot n}$ раз меньше. Следовательно, при одинаковых результатах, он более предпочтителен для решения поставленной задачи. Третий метод имеет более сложную модель (матрицы смежностей и расстояний), которая позволяет находить не только кратчайшие, но всевозможные расстояния от i-й вершины до всех остальных. Поэтому он более универсален по отношению к предыдущим методам, но, вместе с тем, и более трудоёмок, чем эти методы. Для сравнения с ними необходимо применять двухкритериальную оценку качества моделей (см. табл. 3).

Методы вычисления расстояний во взвешенном графе — Таблица 3

NN п/п	Наименование метода (алгоритма)	Универсальность (по числу путей)	Трудоёмкость
1	Умножение на матрицу расстояний	$(n-1)$	$2\cdot n^2\cdot(n-1)$
2	Алгоритм Дейкстра	$(n-1)$	$n\cdot(n-1)$
3	На основе модифицированную матрицу смежности	$(n-1)\cdot(n-1)$	$n\cdot(n^2-1)+n\cdot(n-1)/2$
	Предпочтение	max	min

В последней строке табл. 3 приведены предпочтения на множестве значений показателей. Для указанных предпочтений множество Парето образуют 2-й и 3-й алгоритмы. Для выбора одного из них следует агрегировать численные оценки, приведя их к единой шкале. В качестве нормирующих величин используются оценки 3-го алгоритма, превышающие оценки других алгоритмов. На двухкритериальные оценки алгоритмов влияет важность показателей, назначаемая заказчиком (экспертом).

Заключение

В докладе предложено при решении прикладных задач квалиметрии моделей и полимодельных комплексов использовать алгебраический подход. В основу данного подхода положена модель многосортной алгебраической системы в языке предикатов первого порядка. Разделение сигнатуры модели на отношения и функции позволило сформулировать понятия модели-прототипа и модели метода. В совокупности они составляют модель задачи моделирования. Качество модели предложено измерять относительно решаемой задачи моделирования. Модель-прототип используется для формулирования свойств адекватности, сложности и универсальности, а модель метода – для определения трудоёмкости моделирования. Понятие сортности алгебраической системы использовано для оценивания сложности преобразования информации между взаимодействующими моделями полимодельного комплекса. На основе понятия модели-экземпляра формулируется свойство достоверности результата моделирования, конкретизируемой по отношению к классам решаемых задач. Для измерения перечисленных свойств модели предложен перечень показателей, которые подлежат агрегированию для вычисления общей оценки качества модели. Предложен метод многокритериального оценивания моделей и приведён пример применения этого метода.

Данная работа была выполнена при финансовой поддержке РФФИ (гранты №№ 11-08-01016, 11-08-00767, 12-07-13119-офи-м-РЖД, 12-07-00302, 13-07-00279, 13-08-00702, 13-08-01250, 13-01-00912), Программы фундаментальных исследований ОНИТ РАН (проект №2.11), проекта ESTLATRUS 2.1/ELRI –184/2011/14 «Integrated Intelligent Platform for Monitoring the Cross-Border Natural-Technological Systems» (2012–2013 гг.), проекта ESTLATRUS/1.2./ELRI-121/2011/13 «Baltic ICT Platform».

Литература

1. Калинин В.Н., Резников Б.А. Теория систем и управления (структурно-математический подход). – Л.: ВИКИ, 1987.
2. Юсупов Р.М., Иванищев В.В., Костельцев В.И., Суворов А.И. Принципы квалиметрии моделей // IV СПб Международная конференция «Региональная информатика-95», тез. докладов. – СПб, 1995.
3. Ростовцев Ю.Г., Юсупов Р.М. Проблема обеспечения адекватности субъектно-объектного моделирования// Известия ВУЗов. Приборостроение. - № 7, 1991.
4. Андрианов Ю.М., Суббето А. И. Квалиметрия в приборостроении. – Л. Машиностроение, 1990. – 216.
5. Технология системного моделирования / Е.Ф. Аврамчук, А.А. Вавилов, С.В. Емельянов и др.; Под общ. ред. С.В. Емельянова и др. – М.: Машиностроение; Берлин: Техника, 1988.
6. Соколов Б.В., Юсупов Р.М. Концептуальные основы оценивания и анализа качества моделей и полимодальных комплексов // Известия РАН. Теория и системы управления, 2004, №6, стр. 5-16.
7. Охтилев М.Ю., Соколов Б.В., Юсупов Р.М. Интеллектуальные технологии мониторинга и управления структурной динамикой сложных технических объектов. М.: Наука, 2006. 410 с.
8. Карпов Ю. . Имитационное моделирование систем. Введение в моделирование с AnyLogic 5. –СПб.: БХВ-Петербург, 2005, –400с.

9. Соболева Т.С., Чечкин А.В. Дискретная математика. – М.: Академия, 2006, –255 стр.
10. Микони С.В. Многокритериальный выбор на конечном множестве альтернатив. Учебное пособие. – СПб.: Лань, 2009, 272 с.
11. Микони С.В. Дискретная математика для бакалавра: множества, отношения, функции, графы: Учебное пособие. –СПб.: Издательство «Лань», 2012. –192 с.

Опубликовано: Микони С.В., Соколов Б.В., Юсупов Р.М. Применение алгебраического подхода в квалиметрии моделей и полимодельных комплексов // Сборник докладов VI научно-практической конференции «Имитационное моделирование. Теория и практика». –Казань: Изд-во «Фэн», 2013, Том 1, с. 68-79.