

А. Н. Соломатин, канд. физ.-мат. наук, вед. науч. сотр., e-mail: a.n.solomatin@bk.ru,
Федеральный исследовательский центр "Информатика и управление" РАН, Москва

Интегрированные системы регионального программирования и автоматизация их разработки

Рассматриваются особенности интегрированных систем регионального программирования, состоящих из отдельных систем регионального планирования, проектирования, экологической безопасности и т. д. Интегрированная система строится непосредственно под конкретную задачу за счет автоматического синтеза программ-конвертеров, обеспечивающих информационные связи между этими системами, и существует как единое целое только в виде компьютерной модели.

Ключевые слова: интегрированные системы регионального программирования, системы планирования и проектирования, автоматизация разработки, автоматический синтез расчетных программ

Введение

В течение многих лет в отделе методов проектирования развивающихся систем ФИЦ "Информатика и управление" РАН ведется работа по разработке математических моделей, методов, алгоритмов и программных систем для решения задач комплексного освоения территорий. Было сформировано новое научное направление "региональное программирование" — область региональной экономики, где разрабатываются теория и количественные методы решения задач, возникающих при составлении проектов программ комплексного освоения территорий [1].

Задачи комплексного освоения добывающих регионов возникают при формировании и развитии на них территориально-производственных комплексов в целях освоения природных ресурсов этих территорий. Общая задача комплексного освоения разбивается на ряд тесно связанных друг с другом частных задач, таких как планирование развития основных и сопутствующих отраслей, формирование проектов развития этих отраслей, формирование схем развития производственной и социальной инфраструктуры, вариантов расселения, определение воздействия на экосистему региона и т. д.

Решение перечисленных задач комплексного освоения обеспечивается с помощью систем регионального программирования (Р-систем) —

специализированных автоматизированных систем различного назначения, в частности, систем планирования, проектирования и управления для различных отраслей, экологического мониторинга, оценки экологических рисков и т. д. [2].

Эти системы обладают многими специфическими особенностями, поэтому их можно рассматривать как самостоятельный класс сложных автоматизированных систем со следующими особенностями [3]:

- сложность объекта автоматизации, которым является осваиваемая территория со всем разнообразием природных, экономических и социальных характеристик и связей;
- необходимость комплексного решения задач планирования, проектирования, управления для различных отраслей и уровней управления;
- необходимость увязки во времени и пространстве программ развития различных отраслей;
- сложность и разнообразие используемого математического аппарата из различных областей математики;
- невозможность полной формализации описания как объекта освоения, так и процесса выбора проектов;
- неограниченно долгое время жизни развивающегося объекта комплексного освоения.

1. Интегрированные системы регионального программирования

ИР-системы и принципы их разработки. При решении задач комплексного освоения территорий могут одновременно использоваться сразу несколько различных Р-систем, образуя в процессе решения задачи некоторую единую систему. Так, в работе [2] приводится пример совместного использования нескольких Р-систем при решении задач планирования и проектирования развития нефтегазодобывающих районов Западной Сибири. Будем называть такие комплексы систем интегрированными системами регионального программирования (ИР-системами). Очевидно, что для различных задач должны быть сформированы различные ИР-системы. Для автоматизации разработки и сопровождения ИР-систем [2, 3] предлагается использовать средства автоматического синтеза программ [4].

Предлагаются следующие основные принципы разработки ИР-систем [2, 3, 5], важнейшими из которых мы считаем принципы самоорганизации, децентрализации, виртуальности и конкретности:

- модульность — ИР-системы формируются за счет различной компоновки отдельных Р-систем, которые разрабатываются и могут функционировать независимо друг от друга;
- интегрированность — ИР-системы обеспечивают интеграцию разнородных составляющих систем Р-систем одновременно по нескольким направлениям — по уровням управления, по отраслям экономики, в территориальном аспекте, по решаемым задачам и т. д. с возникновением синергетического эффекта;
- открытость — ИР-системы являются открытыми системами, ориентированными на длительный процесс постоянного развития, когда постоянно изменяются объекты планирования и проектирования, операционная среда, решаемые задачи и т. д., что влечет за собой изменение состава и функций составляющих Р-систем;
- интеллектуальность — непрерывное развитие ИР-систем на основе подходов искусственного интеллекта, включая различные способы представления знаний и автоматический синтез программ;
- самоорганизация — автоматическое построение и развитие ИР-систем как самоорганизующихся и адаптивных систем с заранее неизвестными свойствами, которые без участия человека могут формировать и изменять свою структуру и функции при

изменении внешней среды на основе использования средств автоматического синтеза структур ИР-систем, специальных программ-конвертеров (преобразователей) показателей и планирования вычислений;

- децентрализация — использование ориентированной на данные парадигмы к созданию сложных систем, которые строятся из независимых Р-систем-"кирпичиков" на основе анализа потоков данных и установления связей между этими системами за счет автоматического синтеза программ-конвертеров показателей; это согласуется с принципами организации сложных биологических и социальных систем, неограниченная сложность которых достигается за счет компоновки ограниченного числа элементов при отсутствии общих спецификаций всей системы;
- виртуальность — разрабатываемые ИР-системы существуют как единое целое лишь в виде информационной компьютерной модели; функционирование такой системы сводится к вызову отдельных Р-систем в соответствии с моделью;
- конкретность — построение ИР-систем непосредственно под конкретную задачу на основе использования, как уже говорилось, средств автоматического синтеза структур, программ-конвертеров и планирования вычислений;
- жизненный цикл — поддержка полного жизненного цикла ИР-систем (разработка, функционирование, сопровождение, утилизация) как единого сквозного процесса непрерывного развития, в котором чередуются этапы постепенного совершенствования системы и ее функционирования;
- программирование систем — адаптация всего арсенала методов и средств, относящихся к отдельным программам (синтез программ, схемы программ, планирование вычислений, повторное использование программ и т. д.) при разработке сложных комплексов систем;
- оптимизация — во всех необходимых случаях формируется не одна, а множество "близких" альтернатив решения конкретной задачи, из которых выбирается альтернатива, наилучшая по многим критериям оценки;
- прототипность — ИР-системы должны служить основой для создания на их базе семейств систем для других объектов и задач как за счет модификации базовых ИР-систем в направлении построения других систем, так и за счет использования "пустых" ИР-систем-оболочек, состоящих лишь из инвариантных компонентов;

- распределенность — построение ИР-систем с ориентацией на функционирование в распределенной вычислительной среде с учетом топологии и характеристик этой среды.

Инструментальная система ИРИС. Поддержку полного жизненного цикла интегрированных систем, включая разработку, функционирование и развитие, предлагается выполнять на основе интеллектуальной инструментальной системы ИРИС (Инструментарий Разработки Интегрированных Систем) [2, 3, 5].

Комплекс "ИР-система — ИРИС" работает следующим образом. Развитие осваиваемого региона изменяет состояние среды ИР-системы, а воздействия среды подаются на вход ИРИС, формируя задачу функционирования системы. Используя модели базы знаний и спецификацию задачи, ИРИС строит (изменяет) соответствующую информационную графовую модель ИР-системы (см. п. 2) и/или план решения конкретной задачи. Построенная интегрированная система решает соответствующие задачи регионального программирования, что влечет в будущем изменение среды осваиваемого региона и т. д. Таким образом, ИРИС выступает в качестве адаптирующего элемента ИР-системы, включенного в обратную связь между интегрированной системой и моделируемой средой — региональной социально-экономической системой.

В целом ИРИС решает три основные задачи: автоматизации построения ИР-системы, автоматизации сопровождения и повторного использования ИР-системы, автоматизации управления функционированием ИР-системы; при этом первые две задачи фактически сливаются в единую задачу автоматизации непрерывного развития интегрированной системы.

В комплексе "ИР-система—ИРИС" можно выделить следующие взаимодействующие компоненты [2, 3]:

- основные методологические принципы разработки и функционирования ИР-системы ("принципы");
- этапы жизненного цикла ИР-системы, поддерживаемые ИРИС ("этапы");
- методы автоматического синтеза программ, используемые в процессах ИРИС ("методы");
- входная информация для функционирования ИР-системы ("входы");
- лица, осуществляющие разработку, эксплуатацию и сопровождение ИР-системы ("пользователи");
- информационные модели ИР-системы, обрабатываемые ИРИС ("модели");
- процессы обработки информационных моделей ИР-системы ("процессы");

- база знаний, описывающая предметную область, процесс и средства развития ИР-системы ("база знаний");
- средства выбора вариантов в ситуациях неоднозначности ("выбор");
- технологические процессы обработки и анализа полученных моделей ИР-системы ("технологии");
- различные свойства разрабатываемых ИР-систем ("свойства");
- вычислительные средства, на которых работают Р-системы интегрированной системы ("сеть").

Для работы ИРИС не требуется детальных спецификаций разрабатываемых ИР-систем, поскольку:

- развивающаяся система не может быть полностью специфицирована заранее, причем детальные спецификации вступают в противоречие с растущей сложностью систем;
- возможности самоорганизации и адаптации уменьшают актуальность детальных спецификаций.

Инициализация работы ИРИС может осуществляться либо вручную либо автоматически специальным блоком слежения при появлении во внешней среде или во внутренней модели ИРИС различных изменений, которые касаются параметров внешней среды, целей функционирования, множеств входных и выходных показателей ИР-системы, ограничений на предметную область, характеристик вычислительной среды и т. д.

Модели систем. В основе функционирования ИРИС лежит обработка информационных моделей интегрированных систем — графовых моделей систем (ГМС) [2, 3]. Каждая ГМС представляет собой ориентированный граф $G(S, R)$, где вершинам из множества S (элементам ГМС) в общем случае соответствуют Р-системы интегрированной системы, а дугам из множества R — их информационные связи, реализуемые в виде специальных программ-преобразователей технико-экономических показателей (далее конвертеров).

Компонентами ГМС являются элементы, конвертеры, предикаты и указатели на другие ГМС. Если некоторая Р-система s' использует выходные данные Р-системы s'' , $s', s'' \in S$, то от s' к s'' проведена дуга $r \in R$, которой соответствует программа-конвертер, преобразующая показатели из формы для s' в форму для s'' . Каждому элементу ГМС может быть сопоставлен также некоторый предикат, задающий условия его работы.

Несмотря на простоту формальной модели ГМС являются достаточно сложными объектами.

- ◆ Элементы ГМС могут иметь различную содержательную интерпретацию: отдельный программный модуль, программа, Р-система, ссылка на другую ГМС, а также метаэлемент.
- ◆ ГМС могут образовывать многоуровневые иерархические структуры при наличии в них элементов, ссылающихся на другие ГМС.
- ◆ Каждый метаэлемент задает семейство близких в некотором смысле элементов, которые могут быть получены при его конкретизации. Возможны различные способы задания метаэлементов, такие как место для вставки конкретного элемента, ограничения пользователя, элемент-ссылка на некоторую модель, условная конструкция, набор альтернатив и т. д.
- ◆ При наличии в ГМС метаэлементов такие модели систем превращаются в метамодели, описывающие целые семейства близких в некотором смысле ИР-систем.
- ◆ Элементы ГМС могут существовать в различных вариантах (версиях), различающихся, например, особенностями реализации.
- ◆ ГМС могут иметь различные способы представления: фрейм в базе знаний, визуальное представление на экране монитора, текстовое представление на простом языке программирования ГМС, а также представление в виде сети Петри, используемое при управлении работой ИР-систем в распределенной вычислительной среде.

Отдельные Р-системы, составляющие ГМС, могут быть классифицированы с точки зрения следующих свойств:

- общесистемные свойства, такие как стохастичность, динамичность, нестационарность, адаптивность и т. д.;
- пользовательские свойства, задающие классификацию систем по различным направлениям: по типу объекта ИР-системы, в функциональном аспекте (планирование, проектирование, управление и т. д.), по уровням управления, по отраслям, в территориальном и временном аспекте, по характеру решаемых задач и т. д.;
- характеристики реализации, такие как язык программирования, требуемая память, адрес в сети Интернет и т. д.;
- характеристики ГМС как графов, например, цикломатическое число.

Каждый элемент ГМС (Р-система) может быть представлен как точка в n -мерном метрическом пространстве классификационных признаков, где расстояние между точками-системами есть число различающихся значений этих признаков. Это дает возможность исполь-

зовать такие понятия, как расстояние между системами, окрестность системы, базовые системы (прототипы) и т. д. При этом планарный граф ИР-системы $G(S, R)$ может быть заменен на изоморфный и более информативный граф в n -мерном пространстве классификационных признаков, вершины которого соответствуют точкам этого пространства.

Каждый элемент ГМС описывается своим фреймом-паспортом, который содержит, в частности, следующую информацию: тип элемента (программа, Р-система, ссылка на другую ГМС, метаэлемент), входные и выходные данные системы, ссылки на соответствующие программы-конвертеры, предикат применимости, ролевая характеристика элемента, значения пользовательских и общесистемных свойств, характеристики реализации, местонахождение системы в вычислительной среде, характеристики визуализации на экране монитора.

Эти паспорта хранятся в базе знаний (БЗ) системы ИРИС наряду с другими компонентами, описывающими цели функционирования ИР-системы, параметры среды, входные и выходные показатели, конвертеры, предикаты, планы (цепочки) вычислений, вычислительную среду, вычислительные узлы и т. д. Всего БЗ содержит около 40 компонентов, организованных в виде сети фреймов.

Процессы ИРИС. ИРИС функционирует на основе сквозной технологии обработки графовых моделей и других знаний о разрабатываемой ИР-системе с использованием процессной модели [2, 3]: функционирование ИРИС представляет собой совокупность различных взаимодействующих процессов обработки ГМС, обменивающихся сообщениями. Процессы ИРИС разделяются на следующие группы (блоки).

1. Интерфейсный блок обеспечивает взаимодействие ИРИС с пользователем и внешней средой:

- специфицирование — ввод, анализ и сохранение поступающих на вход ИРИС спецификаций задач;
- редактирование — формирование, просмотр и изменение фреймов-паспортов компонентов БЗ;
- визуализация — анализ и редактирование визуальных представлений ГМС;
- мониторинг — анализ и распознавание сигналов среды для поддержки процесса адаптации ИР-систем.

2. Блок обработки моделей проводит различные операции анализа и обработки ГМС:

- операции с моделями — объединение, пересечение, редукция, выделение и конкретизация ГМС;

- поиск — поиск в базе знаний ГМС, удовлетворяющих заданным ограничениям;
- верификация — проверка полноты и корректности связей в ГМС;
- графовый анализ — получение характеристик ГМС как графов;
- документирование — выдача документации по ИР-системе.

3. Блок настройки моделей обеспечивает настройку ГМС на конкретную задачу:

- синтез конвертеров — автоматический синтез программ-конвертеров с помощью средств автоматического синтеза расчетных программ [4];
- планирование вычислений — автоматическое выделение из ГМС плана вычислений (цепочки вызовов Р-систем и программ-конвертеров) для решения конкретной задачи.

4. Блок исполнения обеспечивает функционирование построенной ИР-системы, в том числе в распределенной вычислительной среде:

- исполнение — интерпретация плана вычислений, т. е. собственно функционирование ИР-системы;
- синхронизацию — динамическое назначение Р-системам и конвертерам вычислительных узлов при распараллеливании работы ИР-системы в распределенной вычислительной среде.

5. Блок процессов развития: семь процессов этого блока более подробно рассмотрены ниже в п. 3.

В процессе планирования вычислений автоматически строится план решения конкретной задачи нахождения целевых показателей работы ИР-системы V по заданным показателям U ; этот план представляет собой линейную последовательность Р-систем и конвертеров, упорядоченных в порядке их вызова. Из имеющейся ГМС выделяется подграф для решения этой задачи, т. е. проводится редукция тех элементов ГМС и конвертеров, которые не используются для решения данной задачи. Планирование на ГМС выполняется аналогично планированию на вычислительных моделях [6] и может осуществляться либо "снизу вверх" от U к V , либо "сверху вниз" от V к U . В процессе планирования также могут проверяться и сниматься предикаты применимости, зависящие от параметров среды, и использоваться ограничения пользователя на вид планов.

Блок исполнения ИРИС вызывает и анализирует план вычислений, проверяет предикаты применимости Р-систем и осуществляет вызов компонентов ГМС — отдельных Р-систем и программ-конвертеров как ехе-файлов. Функционирование ИР-системы может проводить-

ся в распределенной вычислительной среде, включая многопроцессорные системы, кластеры, территориально-распределенные гетерогенные грид-системы. В этом случае блок исполнения позволяет динамически распределять исполнение компонентов ГМС между свободными вычислительными узлами, запускать компоненты одного уровня плана параллельно на различных узлах и синхронизировать исполнение компонентов.

2. Автоматический синтез расчетных программ

Технико-экономические показатели и их преобразования. Для построения программ-конвертеров в процессе разработки ИР-систем используются средства автоматического синтеза расчетных программ (РП) — программ обработки технико-экономических показателей (ТЭП) [4]. Следует отметить, что РП составляют значительную часть программного обеспечения автоматизированных систем различного назначения, включая Р-системы. Каждый ТЭП характеризуется своим наименованием t и значением \bar{t} (вектор либо матрица чисел). Совокупность ТЭП для данной предметной области можно рассматривать как некоторый информационный язык фасетного типа [7].

Каждый фасет определяет некоторую ролевую характеристику языка ТЭП и задает позицию в тексте ТЭП, которая может быть заполнена некоторым термином из заданного множества значений фасетов. Пример ТЭП (в скобках указаны названия фасетов): "при-рост (способ расчета) плановой (функция) численности (характеристика) работающих (предмет) по годам (разрез) в тыс. (коэффициент) человек (единица измерения)".

Вводится n -мерное семиотическое гиперпространство наименований ТЭП $T = \prod_{i=1}^n T_i$, где T_i — множество значений i -го фасета; n — число фасетов [4]. Тогда каждому ТЭП будет соответствовать точка $t \in T$ с координатами (t^1, t^2, \dots, t^n) , $t^i \in T_i$. Пространство T можно рассматривать как метрическое пространство, где расстояние $d(x, y)$ между двумя ТЭП $x, y \in D$ — число несовпадающих значений фасетов в их наименованиях; это позволяет использовать, в частности, понятия расстояния от точки до множества, шара, связного множества и т. д.

Особенности расчетных программ определяются спецификой языка ТЭП. Каждая РП представляет собой линейную последовательность программных фрагментов (вычислительных преобразований, ВП), которые сво-

дятся к циклическому выполнению поэлементных преобразований матриц — значений ТЭП. Каждому ВП значений ТЭП вида $f: \bar{X} \rightarrow \bar{y}$ соответствует некоторое семантическое преобразование (СП) их наименований $s: X \rightarrow y, X \subseteq D, y \in D$. Вводятся различные классы СП и показываются их взаимные связи [4]. Так, каждому правильному СП из класса R соответствует некоторое ВП; класс E элементарных СП включает СП из R , неразложимые на другие СП из R . Важный класс B базовых СП (БСП) включает СП из E , аргументами которых могут быть только семантически близкие (различающиеся в одном фасете) ТЭП.

Назовем фасеты определяющими для данного СП из E , если они задают семантику преобразования и однозначно определяют вид сопоставленного ВП. Систему фасетов назовем независимой, если любое СП из E имеет единственный определяющий фасет. Можно показать, что для независимых фасетов

$$B \equiv E = \bigcup_{i=1}^n B_i, B_i = \bigcup_{k=1}^{m_i} B_{ik},$$

где эквивалентные БСП B_{ik} изменяют i -й фасет одинаковым образом. Поэтому каждому B_{ik} можно сопоставить единственное базовое преобразование значений фасетов (БФП), заданное на T_i , и единственное базовое ВП (БВП) значений ТЭП, что резко уменьшает число связей ТЭП.

БВП обычно реализуют матричные арифметические и структурные операции над матрицами и векторами. Так, множеству B_{ik} , включающему БСП {"план добычи газа", "объем добычи газа"} \rightarrow "дефицит добычи газа", соответствует БФП {"план", "объем"} \rightarrow "дефицит" и БВП — поэлементное вычитание матриц.

В работе [4] было показано, что зависимые системы фасетов обычно можно преобразовать в независимые, исключив зависимые фасеты, не являющиеся определяющими ни для одного СП из E , или проведя редукцию множеств значений фасетов.

Все сказанное выше позволяет поставить задачу синтеза расчетных программ. Пусть заданы: граф $G(A, R)$ — вычислительная модель, где множеству вершин A соответствует объединение множеств допустимых ТЭП из D и их семантических отношений; входные ТЭП U , их значения \bar{U} и выходные ТЭП $V (U, V \subset D)$. Требуется построить РП $P: \bar{U} \rightarrow \bar{V}$, выделив из соответствующих отношений последовательность операторов для расчета значений \bar{V} по \bar{U} .

Теорема. Пусть для пространства ТЭП фасеты являются независимыми, и пусть U —

множество ТЭП, для которых известны их значения \bar{U} . Тогда семантическое преобразование $s: X \rightarrow y, X \subseteq U, y \in D \setminus U$, есть композиция БСП тогда и только тогда, когда $y \in U^*$, где U^* — множество достижимости для U .

Как следствие, при независимости фасетов семантическая модель программы $S: U \rightarrow V$, соответствующая расчетной программе $P: \bar{U} \rightarrow \bar{V}$, есть композиция БСП, меняющих по очереди значения различных фасетов.

Метод и алгоритмы синтеза расчетных программ. Метод синтеза РП [4] существенным образом использует следующие специфические особенности ТЭП и их преобразований: можно количественно оценить семантическую близость ТЭП, установить соответствие между СП наименований и ВП значений ТЭП, причем аргументами БСП могут быть только семантически близкие ТЭП. А сам метод синтеза РП опирается на комплексное использование идей таких методов синтеза программ, как программирование в потоках данных, композиционное программирование и синтез на вычислительных моделях [6].

Рассмотренные свойства ТЭП и их преобразований позволяют обеспечить автоматический синтез программ расчета показателей. Автоматический синтез РП проводится в два этапа: вначале строится семантическая модель программы, затем по ней генерируется текст программы на конкретном языке программирования. Это позволяет учитывать специфику ТЭП и их преобразований, проводить синтез в терминах формальных объектов, оптимизировать РП и строить их на различных выходных языках.

На основании теоремы решение задачи синтеза семантической модели преобразования $s: X \rightarrow y$ можно искать как линейную последовательность (композицию) абстрактных однородных компонентов — БСП, переходя затем к соответствующим вычислительным реализациям — БВП. Анализируется множество БСП, через которое "плывет" множество наименований ТЭП. Если имеется необходимое число семантически близких ТЭП, являющихся аргументами некоторого БСП, то данное БСП активизируется (вырабатывая наименование нового ТЭП) и добавляется к решению.

Семантическая модель РП строится в виде частично упорядоченной в порядке применения БСП последовательности дескрипторов БСП $\langle X, y, f, k \rangle$, где $f: \bar{X} \rightarrow \bar{y}$, а k — уровень вычислений. Следует отметить, что БВП и БСП представляют собой атомарные элементы РП и их семантики, выделение которых обеспечивает многоуровневую декомпозицию про-

грамм, их семантических моделей и значительно уменьшает сложность исходной задачи синтеза, позволяя искать оптимальное решение.

Алгоритм синтеза семантических моделей программ $S: U \rightarrow V$, оптимальных по сложности вычислительных операций, работает в два этапа [4]. Вначале определяется разрешимость исходной задачи, и строится явный граф G' с вершинами из U_0^* , $V \subseteq U_0^* \subseteq U^*$, где U^* есть множество достижимости для U ; одновременно формируется семантическая модель РП $S': U \rightarrow U_0^*$. Для поиска решения используется перебор на графе в ширину в прямом направлении [12]. Для каждой вершины z графа G' находится и запоминается оценка стоимости $g(z)$ ее достижения, т. е. оценка поддерева G' с корнем в z . Алгоритм заканчивает работу за линейное время $N = O(|U^*|)$.

На втором этапе из S' выделяется оптимальное решение — модель искомой РП $S: U \rightarrow V$ с минимальной сложностью вычислительных операций. Для этого граф G' просматривается в обратном порядке, от V к U , и на нем выделяется решающий подграф минимальной стоимости G'' , такой что его конечным вершинам соответствуют только ТЭП из V . Алгоритм упорядоченного эвристического перебора на И/ИЛИ графах [8] всегда находит оптимальный решающий подграф, так как для любой вершины z выполняется условие $g(z) \leq g^*(z)$, где $g^*(z)$ — действительная стоимость для z . Работа алгоритма упрощается за счет того, что существование графа G'' следует из разрешимости задачи, и этот граф ищется на заранее построенном графе G' с известными оценками вершин $g(z)$.

На этапе генерации формируется текст расчетной программы на заданном языке программирования. Вначале по семантической модели РП, сведениям о предметной области и задаче синтеза строится детальная модель РП в виде совокупности дескрипторов основных ее компонентов — типов размерности, переменных, массивов, операторов обмена с информационной базой, циклов, расчетных формул и т. д. Одновременно проводится частичная оптимизация генерируемой РП по памяти и быстрдействию — минимизируется число обменов с базой знаний и циклов.

На втором этапе генерации непосредственно строится текст РП на некотором подмножестве языка программирования блочного типа; сведения о языке задаются правилами вывода контекстно-свободной грамматики языка. В процессе генерации при раскрытии правил вывода срабатывают сопоставленные им семантические правила, которые, извлекая ин-

формацию из детальной модели программы и базы знаний, формируют значения семантических атрибутов, определяющих конкретный вид порождаемой РП (включая идентификаторы, имена файлов, вид БВП и т. д.).

Рассмотренные методы и алгоритмы были реализованы в Системе автоматического синтеза расчетных программ (САС/РП) [4], которая позволяет автоматически формировать программы расчета технико-экономических показателей на заданном языке программирования. Для постановки задачи синтеза требуется минимальная информация — тексты входных и выходных показателей расчетной программы.

3. Автоматизация разработки интегрированных систем

Синтез программ-конвертеров. Автоматизация разработки и сопровождения ИР-систем проводится средствами ИРИС. При построении ИР-систем из независимых Р-систем следует обеспечить информационные связи между их входами и выходами. В различных системах данные с одинаковой семантикой могут представляться по-разному, поэтому для каждой пары систем s_1 и s_2 таких, что $In(s_2) \cap Out(s_1) \neq \emptyset$, необходима программа-конвертер, преобразующая часть выходных данных $Out(s_1)$ системы s_1 (из формы для s_1) во входы $In(s_2)$ системы s_2 (в форму для s_2) [2, 5].

Для автоматического синтеза таких программ необходимо, чтобы обрабатываемыми данными были технико-экономические показатели с единым словарем значений фасетов. Назовем ТЭП x_1 и x_2 семантически эквивалентными, $x_1 \approx x_2$, если их наименования имеют одинаковые значения для определяющих семантику ТЭП фасетов, хотя могут иметь различные значения в служебных фасетах, таких как "функция", "разрез", "единица измерения" и т. д.

Для каждой пары Р-систем s_1 и s_2 программа-конвертер есть совокупность вычислительных преобразований показателей $f: \bar{x}_1 \rightarrow \bar{x}_2$ для каждой пары ТЭП x_1 и x_2 , таких что $x_1 \approx x_2$, $x_1 \in Out(s_1)$, $x_2 \in In(s_2)$. Тогда при использовании автоматического синтеза программ [4] каждое такое преобразование f строится как композиция БВП, а соответствующая семантическая модель $s: x_1 \rightarrow x_2$ — как композиция БСП, таких что каждое из них меняет значение некоторого единственного фасета с номером j в наименованиях ТЭП x_1 и x_2 .

Таким образом, автоматическое построение конвертеров обеспечивает интеграцию разно-

родных Р-систем в единое целое и поддержку постоянного развития (самоорганизации) сложных ИР-систем.

Процессы развития интегрированной системы.

Процессы развития реализуются через процессы обработки ГМС и других компонентов базы знаний средствами ИРИС для решения задач построения и сопровождения развивающихся ИР-систем. Рассматриваются следующие процессы развития ИР-систем [2, 3].

Автоматический синтез моделей обеспечивает автоматическое построение структуры ГМС за счет установления связей по данным между несвязными элементами множества Р-систем S , т. е. фактически обеспечивает самоорганизацию ИР-системы. Автоматический синтез конвертеров проводится на основе анализа семантики наименований показателей, обрабатываемых Р-системами из S , следующим образом:

- для каждой пары Р-систем $s_1, s_2 \in S$ попарно сравниваются все ТЭП из $Out(s_1)$ и $In(s_2)$;
- для каждого ТЭП $x_1 \in Out(s_1), x_2 \in In(s_2)$ пофасетно сравниваются их наименования для проверки $x_1 \approx x_2$;
- при $x_1 \approx x_2$ информация о получении x_2 по x_1 добавляется к модели конвертера I для систем s_1 и s_2 ;
- если после окончания всех сравнений модель I не пуста, то s_1 и s_2 соединяются дугой на ГМС и выполняется автоматический синтез расчетной программы-конвертера, обеспечивающей преобразование показателей из формы системы s_1 в форму системы s_2 .

Если $|S| = n$, а среднее число входов/выходов систем из S есть m , то требуется провести $O(n^2m^2)$ сравнений наименований показателей. Поэтому для сокращения перебора могут использоваться роли Р-систем в ИР-системе, ограничения пользователя, информация самообучения и т. д.

Автоматическая модификация моделей выполняется при сопровождении, развитии, адаптации ИР-систем к изменениям среды и новым задачам, а также при использовании ИР-систем в качестве прототипов при разработке новых систем. При модификации ГМС могут изменяться состав, связи, характеристики и интерпретация компонентов модели и обрабатываемых данных. Автоматическая модификация ГМС сводится к их повторному автоматическому синтезу и инициируется автоматически резидентным блоком. Этот блок работает в режиме слежения и периодически сравнивает текущее состояние входов ИРИС с тем состоянием, при котором была построена модифицируемая модель.

Диалоговое редактирование моделей обеспечивает построение/модификацию ГМС с использованием средств машинной графики. Пользователь в диалоговом режиме может добавлять, заменять и удалять компоненты ГМС, просматривать их характеристики, объединять, разделять ГМС, формировать метамоделли и т. д.

Задание текстовых описаний моделей может проводиться на простом языке программирования ГМС декларативного типа. Для метаэлементов могут быть заданы ограничения на класс элементов, условные выражения и наборы альтернатив.

Обобщение моделей обеспечивает получение из нескольких ГМС модели более общего вида (метамоделли), которая в будущем может быть использована в качестве базы (прототипа) при разработке новых моделей и модификации уже существующих. После формирования метамоделей исходные модели можно удалить.

Конкретизация метамоделей (моделей с метаэлементами) позволяет строить семейства ГМС с идентичной структурой за счет конкретизации метаэлементов до элементов. Конкретизация каждого метаэлемента может быть задана пользователем, выбрана им среди нескольких вариантов, получена автоматически в зависимости от параметров среды и т. д.

Утилизация моделей обеспечивает повторное использование ранее созданных ГМС, их фрагментов (подграфов) и компонентов (конвертеров, предикатов) при создании моделей новых ИР-систем. Проблемы повторного использования во многом снимаются при использовании формализованных описаний ГМС, что обеспечивает однозначную идентификацию, эффективное хранение и поиск повторно используемых фрагментов.

Заключение

Особенность предложенного подхода состоит в том, что он обеспечивает самоорганизацию и адаптацию сложных программных комплексов, состоящих из независимых систем за счет автоматического синтеза программ-конвертеров, обеспечивающих информационные связи между этими системами. Интегрированная система строится непосредственно под конкретную задачу и существует как единое целое только виртуально, в виде компьютерной модели, что снижает трудоемкость разработки ИР-системы и повышает эффективность ее функционирования. Полученные результаты могут быть использованы не только при раз-

работке ИР-систем, но и при создании производственных комплексов систем, обрабатывающих технико-экономические показатели.

Список литературы

1. Хачатуров В. Р. Математические методы регионального программирования. М.: Наука, 1989. 304 с.
2. Хачатуров В. Р., Соломатин А. Н., Злотов А. В. и др. Планирование и проектирование освоения нефтегазодобывающих регионов и месторождений: Математические модели, методы, применение / Под ред. В. Р. Хачатурова. М.: УРСС: ЛЕНАНД, 2015. 304 с.
3. Соломатин А. Н. Автоматизация разработки интегрированных региональных систем. М.: ВЦ РАН, 1993. 32 с.

4. Соломатин А. Н. Автоматический синтез расчетных программ на основе анализа семантики показателей // Кибернетика и системный анализ. 1994. № 5. С. 133–142.
5. Solomatin A. N. Integrated systems of regional programming // Tenth International Conference "Management Of Large-Scale System Development" (MLSD) / Moscow (October 2017). IEEE Conference Publications, IEEE Xplore Digital Library. URL: <http://ieeexplore.ieee.org/document/8109689> (дата обращения 12.08.2019).
6. Тыгу Э. Х. Решение задач на вычислительных моделях // ЖВМ и МФ. 1970. Т. 10, № 5. С. 716–734.
7. Главатская Н. Г., Петухова О. В., Ясин Е. Г. Базовый информационный язык — система формализованного описания экономических данных // Прикладная информатика. М.: Финансы и статистика, 1983. С. 35–48.
8. Нильсон Н. Проблемы искусственного интеллекта. М.: Радио и связь, 1985. 280 с.

A. N. Solomatin, Ph.D., Leading Researcher, e-mail: a.n.solomatin@bk.ru,
Federal Research Center "Computer Science and Control" of Russian Academy of Sciences,
Moscow, Russian Federation

The Integrated Systems of Regional Programming and Automation of their Development

Features and automation of development of the integrated systems of regional programming (IR-systems) are considered. IR-systems are the complexes of the systems consisting of separate independent systems (R-systems) of regional planning, design, ecological safety and intended for solving problems of complex development of territories. Each integrated system exists as a whole only virtually in the form of a computer model and is built each time for a specific task by automatically establishing information links between P-systems. Support for the full life cycle of IR-systems, including the development and operation is provided by the IRIS tool system. Models of IR-systems processed by IRIS and different groups of IRIS processes (operating on the basis of the process model) are described. The concept, method, and algorithms of automatic synthesis of calculation programs — programs for calculation of technical and economic indicators are discussed. The hyperspace of names of indicators is introduced, various classes of semantic transformations of names are introduced and investigated. The method and algorithms of synthesis of semantic models of programs as a sequence of semantic transformations of special form are described, generation of program text on some subset of a block-type programming language is considered. The issues of automation of the development of IR-systems using IRIS tools are discussed. To build a model of an IR-system for solving a specific problem by means of existing P-systems, information links between systems are automatically established based on the analysis of the semantics of the processed indicators. And then, using a system of the automatic synthesis of calculation programs converter programs are generated that implement these relationships and transform semantically equivalent indicators of P-systems from one form of presentation to another. The development processes of IR-systems implemented by IRIS are listed, including processes of automatic synthesis and modification of models of IR-systems, generalization, utilization of models, etc. The proposed approach can be used to automate the development of fairly arbitrary systems processing technical and economic indicators.

Keywords: complex development of territories, integrated systems of regional programming, planning and design systems, systems complex model, process model of functioning, planning of calculations, distributed functioning environment, automatic synthesis of calculation programs, indicator converters, development automation tools

DOI: 10.17587/it.26.107-115

References

1. Khachaturov V. R. Mathematical methods of regional programming, Moscow, Nauka, 1989, 304 p. (in Russian).
2. Khachaturov V. R., Solomatin A. N., Zlotov A. V. et al. Planning and designing the development oil and gas production regions and fields: Mathematical methods, models, applications, Moscow, URSS, LENAND, 2015, 304 p. (in Russian).
3. Solomatin A. N. Automation of development of the integrated regional systems, Moscow, Published CC RAS, 1993, 32 p. (in Russian).
4. Solomatin A. N. Automatic synthesis of calculation programs based on the analysis of semantics of indicators, *Cybernetics and System Analysis*, 1994, no. 5, pp. 133–142 (in Russian).

5. Solomatin A. N. Integrated systems of regional programming, *Tenth International Conference "Management Of Large-Scale System Development" (MLSD) / Moscow (October 2017)*, IEEE Conference Publications, IEEE Xplore Digital Library, available at: <http://ieeexplore.ieee.org/document/8109689> (date of access: 12.08.2019).
6. Tygu E. H. The solution of problems on computational models, *ZhVM and MF*, 1970, vol. 10, no. 5, pp. 716–734 (in Russian).
7. Glavatskaya N. G., Petukhova O. V., Yasin E. G. Basic information language — the system of formalized description of economic data, *Applied Informatics*, Moscow, Finance and statistics, 1983, pp. 35–48 (in Russian).
8. Nielson N. Problems of artificial intelligence, Moscow, Radio and communication, 1985, 280 p. (in Russian).