

**ПРАКТИЧЕСКИЙ ОПЫТ РАЗРАБОТКИ АЛГОРИТМИЧЕСКИХ МОДЕЛЕЙ****В. Е. Марлей (Санкт-Петербург)**

Автор имеет почти 30-летний опыт работы с конечными пользователями по разработке и практическому внедрению моделей в различных предметных областях. Пользователи, с которыми приходилось взаимодействовать, в основной массе, не имели хорошей математической подготовки и/или были недостаточно компьютерно грамотны. Анализ этого опыта позволяет сделать некоторые предположения об общих для данной группы пользователей закономерностях восприятия проблем.

Познание всегда начинается с дискретизации анализируемого процесса. Фиксируется состояние интересующей части мира, затем производится воздействие или фиксируется какое-либо производимое в это время воздействие и, затем, фиксируется новое состояние. Воздействие фиксируется как причина изменения состояния, новое состояние как следствие воздействия. Восприятие всегда идет конечными этапами. Процесс представляется как некоторая циклически повторяющаяся регулярная процедура со своим “естественным” параллелизмом. Человек рассматривает процесс в естественной дискретности, не определяемой требованиями какого-либо метода вычислений. Если он наблюдает и воспринимает процесс с дискретом сутки, то и модель, сложившаяся в его мозгу, будет иметь дискрет сутки и предсказывать состояние мира на следующие сутки и т.п. Доверие к реализуемой модели повышается по мере включения в нее все большей части известных ранее из практического опыта закономерностей в виде, который позволяет пользователю распознать эти закономерности в модели. Окончательно особенности восприятия для анализируемого класса пользователей можно сформулировать следующим образом:

1. дискретность описания и восприятия анализируемого процесса, стремление рассматривать процесс с шагом, равным привычному дискрету наблюдения и/или съема информации;
2. стремление представить анализируемый процесс как некоторый связный сценарий, с описанием пред- и пост условий выполнения каждого этапа;
3. стремление свести описание к некоторой регулярной циклически повторяющейся процедуре;
4. стремление рассматривать этапы сценария во временной последовательности;
5. стихийный параллелизм описания сценария;
6. стремление уменьшить число используемых единиц измерения и анализируемых показателей, нежелание использовать непривычные единицы измерения и показатели;
7. в большинстве случаев довольно четкое представление о допустимых и оптимальных границах значений анализируемых показателей;
8. недоверие к результатам, полученным методами, которые они не могут наложить на свой сценарий процесса;
9. частое стихийное сведение анализируемого процесса к описанию взаимодействия некоторых “хранилищ”, “емкостей”;
10. стремление рассматривать процесс по иерархически упорядоченным слоям и вместе с тем слабая упорядоченность описания сценария.

С целью снижения барьера между ЭВМ и пользователем необходимо учитывать выявленные закономерности при выборе и создании средств представления моделей для систем автоматизации моделирования. Таким образом, будем ориентироваться на способ представления моделей позволяющий описать моделируемый процесс как регу-

лярную циклически повторяющуюся процедуру, описывающую процесс с учетом дискретности его восприятия и естественного параллелизма и позволяющую пользователю сопоставить вычислительную структуру модели со структурой причинно-следственных связей между явлениями в имеющемся у него сценарии.

Известно два основных подхода в моделировании: функциональный и алгоритмический [1-4]. При функциональном подходе главное, чтобы реакция модели на подобное воздействие была подобна реакции объекта с заданной степенью точности. Соответствие структуры модели и объекта не обязательно. При алгоритмическом подходе модель – это некоторое формальное представление сценария моделируемого процесса, которое имеется у предметного специалиста и ее структура подобна структуре причинно-следственных связей между явлениями, описанными в сценарии. Основным средством достижения адекватности при функциональном подходе является подбор модели и ее идентификация путем нахождения требуемых параметров. Основным средством достижения адекватности для алгоритмической модели является коррекция ее структуры на основе уточнения исходного сценария. Очевидно, что алгоритмический подход по самой своей сути обеспечивает большую степень доверия у пользователей, и алгоритмические модели будут ими легче восприниматься, что и наблюдалось на практике. Под алгоритмической моделью понимается формализованное описание сценария предметного специалиста для моделируемого процесса, структура которого сопоставима со структурой причинно-следственных и временных зависимостей между явлениями моделируемого процесса вместе со всей информацией, необходимой для ее программной реализации.

Алгоритмические модели требуют специальных средств представления, обеспечивающих выполнение указанных выше требований, в частности – параллельными граф-схемами алгоритмов (ПГСА). В данном случае ПГСА будет максимально распараллелена, и ее операционные связи будут изоморфны информационным. В Санкт-Петербургском институте информатики и автоматизации РАН был разработан язык представления структуры алгоритмических моделей на основе “алгоритмических сетей” [5], который изоморфен некоторому специальному виду ПГСА и основан на задании структуры информационных связей между процедурами, представляющими отдельные явления моделируемого процесса.

Алгоритмические сети (АС) используются для представления структуры алгоритмических моделей и являются отображением причинно-следственных и временных связей между явлениями, представленными в сценарии моделируемого процесса, в вычислительные связи между операторами. Таким образом, алгоритмические сети задают некоторую расчетную схему, по которой проводятся вычислительные эксперименты над моделями. Алгоритмические сети – это конечный ориентированный нагруженный граф, вершинам которого сопоставлены операторы или ссылки на другие модели, а дугам – переменные, связываемые операторами. Для задания исходного состояния моделируемого процесса и для описания перехода моделируемого процесса через шаг моделирования служат операторы задержки. АС представляет структуру вычислительной схемы соответствующей причинно-следственным связям для одного шага циклически повторяющегося, дискретно рассматриваемого процесса, с учетом его естественного параллелизма. Требуемое число шагов моделирования, для которых нужно проанализировать процесс, задается для АС внешними параметрами. Дуга в АС означает, что операторы в связываемых ею вершинах вычислительно связаны. Вычисления на АС производятся по шагам моделирования, происходят в каждой вершине сразу же, когда для текущего шага моделирования становятся известны значения всех переменных соответствующих входным дугам вершины. Переменные, соответствующие входным входным дугам, соответствуют исходным, входным переменным, все остальные перемен-

ные – расчетные. Доказано, что для АС можно построить расчетную программу, когда заданы значения всех переменных, соответствующих входным дугам АС. Типы переменных и операторов в АС произвольны. Структура АС удовлетворяет следующим ограничениям:

- не существует контура, в котором хотя бы одна вершина не содержала оператор задержки;
- переменная может быть вычислена только в одной вершине;
- в одной АС все операторы задержки определяют рекуррентные соотношения относительно одной и той же величины и с одинаковым шагом ее изменения, и срабатывают одновременно после вычисления всех остальных вершин.

Можно сказать, что АС являются графическим представлением системы рекуррентных выражений от переменных произвольного типа.

АС представляют собой тело цикла, условие останова которого задано в некотором внешнем патерне.

АС обеспечивает возможность своего параллельного и распределенного выполнения. Сам процесс выполнения АС, при отсутствии ограничений на используемые ресурсы ЭВМ, уже является параллельным процессом. На основе представления АС был разработан метод автоматизации распараллеливания алгоритмов и программ [6].

Представление структуры моделей в виде АС в значительной степени соответствует требованиям, выдвинутым на основе перечисленных в начале статьи закономерностей. Дальнейшее развитие АС привело к разработке распределенных АС и АС со ссылками в вершинах на другие модели [5], которые эквивалентны классу структурных (по Дейкстре) алгоритмов. При определенном уровне и условиях декомпозиции IDEF0-диаграмм можно говорить об их эквивалентности АС. На рис. 1 представлена IDEF0-диаграмма и на рис. 2 эквивалентная ей АС.

На основе формализма алгоритмических сетей было разработано несколько версий системы автоматизации моделирования, которые успешно применялись для создания моделей в различных предметных областях (модели годового и пятилетнего планирования для Госплана РСФСР, эколого-экономическая модель озера Севан, модель процесса производства стекла, модель энергоснабжения судна и т.п.). По результатам работы системы и разработанные модели награждались серебряными и бронзовыми медалями ВДНХ. В настоящее время идет разработка новой версии системы автоматизации моделирования КОГНИТРОН, обеспечивающей полное использование всех возможностей АС, т.е. алгоритмическую полноту, произвольность типов переменных и операторов, возможность распределенного моделирования и т.п.

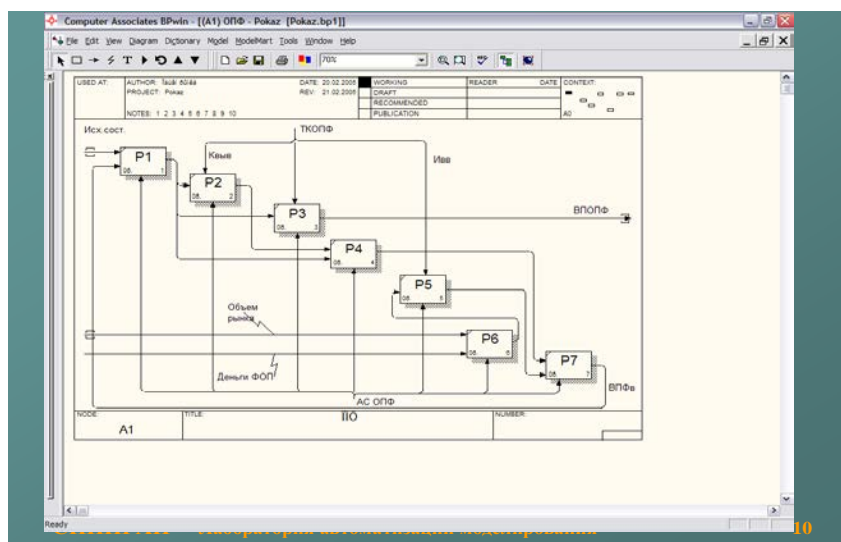


Рис. 1. IDEF0-диаграмма блока экономической модели

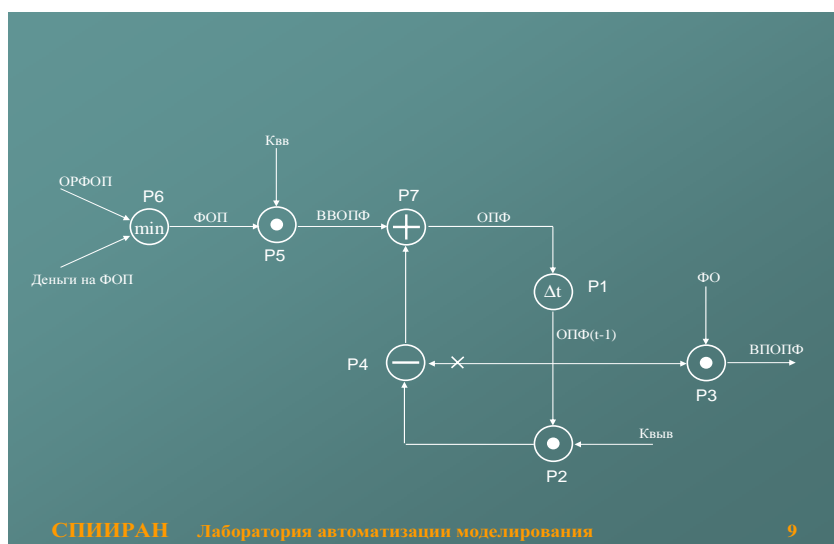


Рис. 2. АС, соответствующая IDEF0-диаграмме на рис. 1

### Литература

1. Бирюков Б.В. Кибернетика и методология науки. – М.: Наука, 1974. – 412 с.
2. Поспелов Г.С., Ириков В.А., Курилов А.Е. Процедуры и алгоритмы формирования комплексных программ. – М.: Наука, 1985. – 423 с.
3. Управление, информация, интеллект/Под ред. Берга А.И. – М.: Мысль, 1976. – 383 с.
4. Пономарев В.М. Алгоритмические модели в задачах исследования систем.// “Алгоритмы и системы автоматизации исследований и проектирования”.. – М.: Наука, 1980. – С. 4-8.
5. Иванищев В.В., Марлей В.Е. Основы теории алгоритмических сетей. – СПб: СПбГТУ, 2000. – 180 с.
6. Марлей В.Е., Воробьев В.И., Крылов Р.А., Петров М.Ю., Ююков Я.А. Автоматизация распараллеливания программ.//Труды СПИИРАН. – СПб: Наука, 2005. – Т. 2. – Вып. 2. – С. 101-110.