Модели потоков работ

Осипов Г. С.
gos@isa.ru
Москва, Институт системного анализа РАН

В наши дни эффективность работы каждой организации связана с эффективностью управления её ресурсами и процессами.

В большинстве случаев действующие организации не имеют исчерпывающих описаний всех реализуемых ими процессов, поэтому на первый план выходят задачи построения таких описаний на основе примеров или, как иногда говорят, рабочих последовательностей.

Каждая рабочая последовательность представима в виде графа, в вершинах которого находятся некоторые работы (операции, мероприятия), а ребра определяют порядок выполнения работ. Как оказалось, рабочие последовательности, даже решающие одну и ту же задачу, могут различаться как порядком выполнения, так и составом работ, поэтому встает задача построения описания всего множества рабочих последовательностей, решающих одну и ту же задачу, т. е. такого графа, что граф каждой рабочей последовательности являлся бы его подграфом.

Для этой цели вводится понятие оператора переходов, т. е., по существу, правила (соответствующего некоторой работе), меняющего состояние процесса. Далее с помощью таких операторов определяются примеры или прецеденты потоков работ, тем самым уточняется понятие рабочей последовательности. Далее строятся описания классов эквивалентности прецедентов. Наконец, описания классов эквивалентности используются для синтеза модели потоков работ, которая, в свою очередь, может служить основой для реинжиниринга бизнес-процессов, оптимизации процессов по различным критериям и т. д.

Потоки работ и процессы

Пусть U — множество слов конечной длины над некоторым алфавитом. Зададим на U семейство алгебраических систем с сигнатурами, включающими одно-, двух- и n-местные отношения на U: $P^1, \ldots, P^m$. Для простоты будем полагать, что в сигнатуры входит ровно по одному отношению каждой местности. Каждую такую алгебраическую систему будем называть *состоянием* и обозначать через $\mathbf{s}$. Множество всех состояний обозначим через $\mathbf{E}$. Элементы многоместных отношений будем далее называть *фактами*, элементы одноместных отношений — *признаками*. Если N — дискретное линейно упорядоченное множество, то семейство отображений $\mathbf{O} = \{o_i\}$, $i = 1, \ldots, M$, $\mathbf{O} \colon \mathbf{E} \times N \to \mathbf{E}$, таких что $o(s,n) = (s, n+1)$, где $(s,n) = \langle (z,n),(p,n) \rangle$ — состояние системы в точке n, $z \subseteq P^i$, $i = 2, \ldots, m$ — множество фактов, $p \subseteq P^1$ — множество признаков, будем называть *множеством операторов переходов*. Далее множество N будем называть временем (дискретным), а для (s,n), (z,n), (p,n) используем более привычные обозначения: $s(n)$ либо s_n и т. д.

Если $\pi \subseteq P^1$, $\varphi \subseteq P^2 \cup \cdots \cup P^m$, то оператор $\mathbf{o} \in O$ имеет вид: $\mathbf{o} = \langle \pi, \varphi \rangle$, и $s(n+1) = \mathbf{o}\, s(n)$, где $\mathbf{o}\, s(n) = \langle z(n+1),\ p(n+1) \rangle$, $z(n+1) = z(n) \cup \varphi$ либо $z(n+1) = \varphi$, $p(n+1) = p(n) \cup \pi$ либо $p(n+1) = \pi$.

Два разных способа действия оператора переходов соответствуют двум различным способам формирования нового состояния: появлению в нем новых фактов и признаков при сохранении имеющихся, либо исчезновению старых признаков и фактов и появлению новых.

Если $\Omega(\mathbf{O})$ — семейство последовательностей вида $\omega = \langle \mathbf{o}_i, \mathbf{o}_j, \ldots, \mathbf{o}_k \rangle$ над множеством $\mathbf{O}$ операторов $\mathbf{o}_j$, где $i, j, \ldots, k$ — элементы множества натуральных чисел $\mathbb{N}$ и $i < j < \ldots < k$, то каждую последовательность ω будем называть *прецедентом* или *примером потока работ*. На $\Omega(\mathbf{O})$ зададим отношение эквивалентности ρ, порождающее фактор-множество Ω_ρ множества $\Omega(\mathbf{O})$.

Описанием $G(\{\omega\})$ каждого класса эквивалентности $\{\omega\} \in \Omega_\rho$ будем называть граф, такой что маршруты, порожденные всеми примерами $\omega \in \{\omega\}$ являются его подграфами.

Классы эквивалентности потоков работ и их построение

Внутри каждого из классов могут оказаться примеры, отличающиеся от других порядком следования операторов, порядком следования их групп, степенью их повторяемости и др. Эти различия приводят к появлению так называемых маршрутов: последовательного, параллельного, конкурентного, итеративного и условного в потоках работ [1] и необходимости представления таких маршрутов в $G(\{\omega\})$. Первые два вида маршрута были описаны в [2]. В клинической медицине, например, в некоторых случаях допускаются различные последовательности лечебных мероприятий для лечения одной нозологической формы.

На множестве матриц инцидентности введем ассоциативную и коммутативную операцию покомпонентного сложения, сохраняющую единицу: если $A = [a_{ij}]$ и $B = [b_{ij}]$ — матрицы инцидентности, то $A + B = [c_{ij}]$, где $c_{ij} = \max\{a_{ij}, b_{ij}\}$.

Пусть $M(\omega_j)$ — матрица инцидентности графа примера ω_j, через $M(G)$ обозначим матрицу инцидентности графа $G(\{\omega\})$.

Теорема 1. *$M(G) = \sum_j M(\omega_j)$, где суммирование в указанном выше смысле выполняется по всем примерам ω_j из класса $\{\omega\}$.*

Эта теорема обосновывает процедуру построения описания класса.

Следующая теорема показывает, что, какой бы пример мы ни взяли, если он принадлежит одному из классов, то он порождает хотя бы один из маршрутов, названных выше.

Теорема 2. *Для любого ω, если $\omega \in \{\omega\}$, то ω порождает хотя бы один из маршрутов в $G(\{\omega\})$.*

Если с каждым оператором связать условие его применимости и состояние, к которому он применяется, то получим понятие процесса над множеством дискретных событий: последовательность $\rho = \big\langle (s_i, \mathbf{c}_i, \mathbf{o}_i), (s_j, \mathbf{c}_j, \mathbf{o}_j), \ldots, (s_k, \mathbf{c}_k, \mathbf{o}_k) \big\rangle$ будем называть *примером процесса*, если для каждых двух её элементов $(s_n, \mathbf{c}_n, \mathbf{o}_n)$ и $(s_{n+1}, \mathbf{c}_{n+1}, \mathbf{o}_{n+1})$ справедливо $\mathbf{c}_{n+1} \in \mathbf{o}_n s_n$, где $\mathbf{o}_n s_n$ — результат применения оператора $\mathbf{o}_n$ к состоянию s_n, а $\mathbf{c}_{n+1}$ — условие применимости оператора $\mathbf{o}_{n+1}$ к состоянию s_{n+1}. Легко видеть, что множество примеров потоков работ изоморфно множеству примеров процессов относительно упорядочения, поэтому приведенные выше утверждения тривиальным образом переносятся на процессы.

Модель потоков работ

Моделью потоков работ будем называть динамическую систему $H = \langle X, N, \Psi \rangle$, где X — дискретное множество событий, N — линейно-упорядоченное дискретное множество, Ψ — функция переходов. Наибольший интерес здесь представляет восстановление функции переходов Ψ, так как множество X определено выше ($X = \mathbf{E}$), а в качестве N можно взять множество натуральных чисел. Что касается Ψ, то $\Psi\colon X \times N \to X$, так что для каждого состояния s_i $\Psi(s_i, n) = s_{i+1}$. Так как в принятом здесь представлении переходы реализуются операторами, то Ψ реализуется следующим алгоритмом:

1. Выбрать оператор, условие которого выполняется в текущем состоянии s_i;
2. Применить оператор, т. е. построить состояние $s_{i+1} = \mathbf{o}\, s_i$,
3. Перейти к пункту 1.

Это означает, что задача восстановления функции Ψ сводится к задаче восстановления операторов и условий их применения. Здесь надо заметить, что каждый из классов $G(\{\omega\})$ может содержать свое множество операторов, поэтому речь должна идти о восстановлении множества операторов каждого из классов и последующем объединении этих множеств. Для восстановления операторов следует для каждого оператора найти π и φ. Для этой цели будем сравнивать пары состояний s_i и s_{i+1}.

Случай 1: $s_i \cap s_{i+1} \neq \varnothing$, тогда $\varphi = z_{i+1} \backslash (z_{i+1} \cap z_i)$, $\pi = p_{i+1} \backslash (p_{i+1} \cap p_i)$. В частных случаях, когда имеет место включение в одну, либо в другую сторону, например $s_i \subseteq s_{i+1}$, получаем $\varphi = z_{i+1} \backslash z_i$, $\pi = p_{i+1} \backslash p_i$.

Случай 2: $s_i \cap s_{i+1} = \varnothing$, тогда $\varphi = z_{i+1}$, $\pi = p_{i+1}$.

Вершины графа $G(\{\omega\})$, в которых начинается любой маршрут, кроме последовательного, будем называть точками ветвления. В них определяются условия применимости операторов. Пусть ρ и ρ' — прецеденты, породившие параллельный маршрут в $G(\{\omega\})$, s^0 — состояние $G(\{\omega\})$ в точке ветвления, s и s' — следующие за s^0 состояния прецедентов ρ и ρ', соответственно. Если говорить неформально, то для определения условий применимости операторов $\mathbf{o}$ и $\mathbf{o}'$ следует сравнить состояния s и s' прецедентов ρ и ρ' и определить их различия, которые в простейшем случае и являются искомыми условиями применимости.

В более сложных случаях условия применимости операторов могут описываться логическими выражениями, например, в языке исчисления предикатов первого порядка. В этом случае условием оператора является конъюнкция атомарных формул языка исчисления предикатов первого порядка, каждая из которых интерпретируется одним из сигнатурных отношений и выполняется на элементах, различающих s и s'.

Заключение

В докладе кратко описан способ построения модели потоков работ по прецедентам. Метод позволяет построить описание технологических или бизнес-процессов сложной системы и, на этой основе, выполнить различные их преобразования, такие как уменьшение различных видов операций, уменьшение количества циклов и структурную оптимизацию.

Работа поддержана грантом РФФИ 06-07-89110.

Литература

[1] *Г. С. Осипов* Обнаружение и исследование потоков работ и процессов над множествами дискретных событий // Труды конференции «Системный анализ и информационные технологии 2005», М.: URSS. — 2005.

[2] *Laura Maruster, A. J. M. M. (Ton) Weijters, Wil M. P. van der Aalst, Antal van den Bosch* A rule-base approach for process discovery. Dealing with noise and imbalance in process logs // Data Mining and Knowledge Discovery, Springer-Verlag. — 2006. — 13. — P. 67–87.