Саратовский государственный университет имени Н. Г. Чернышевского
Институт проблем точной механики и управления РАН
Правительство Саратовской области

КОМПЬЮТЕРНЫЕ НАУКИ И ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Материалы Международной научной конференции

2–3 июля 2018 г.

Саратов

Саратов
Издательский центр «Наука»
2018

на n–ом шаге программы. Аксиомы теории описывают процесс выполнения операторов шаг за шагом посредством интерпретации аксиом вида 2, левая часть которых специфицирует исполняемый оператор, а правая часть — результат его исполнения. Аксиомы интерпретируются на списке $t_{prog} = \widetilde{pr}$, представляющем дерево разбора конкретной программы pr. Приведем спецификацию оператора присваивания:

$$(\forall n)(\forall\, x_{op} \dot{\in} t_{prog})(\forall\, y_{as} \in x_{op})(\forall\, z_{var}, w_{exp} \in y_{as})(Act(y_{as}, n) \rightarrow \\ Val(Name(z_{var}), n) = Vald(w_{exp}, n-1),\, Ter(y_{as}, n),\, Ter(x_{op}, n)).$$

Особенность спецификации условного оператора и оператора цикла связана с проверкой истинности $bexp$. Свойство частичной корректности программы задается формулой: $\varphi(\overline{v}) \rightarrow \psi(\overline{v})$. Здесь φ — конъюнкция тождеств $Val(Name(v_i, 0)) = b_i$ (теории T_0) для значений переменных, вводимых в pr. Формула ψ — конъюнкция тождеств $Vald(Name(v_j, m)) = d_j$, $1 \leqslant j \leqslant k$, для всех значений переменных программы pr, завершившей свое исполнение на шаге m. Эту формулу легко автоматически проверить за время $O(km)$, если сформировать список $\langle\langle id_1, a_1\rangle, \ldots, \langle id_m, a_m\rangle\rangle$ в процессе интерпретации аксиом для as, как значение функции $V(t_{prog})$, состоящее из результатов выполнения всех операторов as. Для верификации параллельных систем выразительных возможностей формул 2 недостаточно. Использование «$\neg$» в левой части и разделительной дизъюнкции в правой части квазитождества позволяет специфицировать альтернативное поведение системы, на модели которой можно верифицировать свойство взаимного исключения [4].

Список литературы

[1] *Goncharov S. S., Ershov Yu. L., Sviridenko D. I.* Semantic programming // Information processing: Proc. IFIP 10th World Comput. Congr. Ser. 10. — Elsevier, North Holland, 1986. — P. 1093–1100.

[2] *Ильичева О. А.* Инициальная семантика логических спецификаций с отрицанием // Кибернетика и системный анализ. — 1992. — №. 4. — С. 13–19.

[3] *Глушкова В. Н.* Гибридные логические модели // Научное программное обеспечение в образовании и научных исследованиях: тр. науч.-техн. конф. — СПб.: Изд-во Политехнического университета, 2008. — С. 21–28.

[4] *Мальцев А. И.* Алгебраические системы. —Москва: Наука, 1970.

Троичное моделирование комбинационных схем с использованием ROBDD

Голубева О. И.

golubeva.to@yandex.ru

Томский государственный университет, Томск, Россия

В работе рассматривается задача троичного моделирования комбинационных схем. Она сводится к задаче вычисления значения интервального расширения булевой функции на интервале, представленном троичным вектором. Предлагается решение, основанное на представлении булевой функции ROBDD графом. Оно позволяет решить задачу за один сокращенный обход графа.

Ключевые слова: троичное моделирование, комбинационная схема, ROBDD, интервальное расширение булевой функции

Введение

Троичное моделирование логических схем может применяться при решении различных проблем диагностирования, например, для построения тестов, при верификации и т. д. В данной работе рассматривается задача троичного моделирования комбинационных схем. Она сводится к задаче вычисления значения интервального расширения булевой функции на интервале, представленном троичным вектором. В работе предлагается решение этой задачи, основанное на представлении булевой функции ROBDD графом. Оно позволяет получить решение за один сокращенный обход графа. В работах [1–3] нами представлен метод троичного моделирования синхронных последовательностных схем, в котором можно использовать предлагаемое решение.

1. Интервальное расширение булевой функции

Для формальной постановки задачи троичного моделирования в работе [4] вводится понятие *интервального расширения булевой функции*.

Обозначим через E^f область определения функции f. Назовем функцию $f(x_1, x_2, \ldots, x_n)$ *полностью определенной булевой функцией*, если $E^f = E_n^2$. Далее будем рассматривать полностью определенную булеву функцию $f(x_1, x_2, \ldots, x_n)$.

Обозначим $M_1^f = \{\gamma \in E_n^2 : f(\gamma) = 1\}$ и $M_0^f = \{\gamma \in E_n^2 : f(\gamma) = 0\}$. M_1^f (M_0^f) — область единичных (нулевых) значений функции f,

$$M_1^f \cup M_0^f = E_n^2.$$

Будем в дальнейшем рассматривать $\times$ как символ, обозначающий любой из элементов множества $\{0,1\}$. Будем называть значения 1 и 0 *определенными*, а значение $\times$ — *неопределенным*.

Вектор $\alpha = (\alpha_0, \ldots, \alpha_n)$, компоненты которого принимают свои значения из множества $\{0,1,\times\}$, будем называть *троичным вектором*.

Троичная функция — это функция $f(x_1, x_2, \ldots, x_n)$, аргументы которой определены на множестве $E^\times = \{0,1,\times\}$, и такая, что $f(\alpha_1, \alpha_2, \ldots, \alpha_n) \in E^\times$, когда $\alpha_i \in E^\times$, $i \in \overline{1,n}$.

Введем на множестве $E^\times = \{0,1,\times\}$ отношение частичного порядка следующим образом: $0 \leqslant \times$ $\times$, $1 \leqslant \times$, 1 и 0 не сравнимы. *Троичные векторы* $\alpha = (\alpha_0, \ldots, \alpha_n)$ *и* $\beta = (\beta_0, \ldots, \beta_n)$ *находятся в отношении* $\alpha \leqslant \beta$, если $\alpha_i \leqslant \beta_i$, $i \in \overline{1,n}$.

Троичный вектор $\alpha = (\alpha_0, \ldots, \alpha_n)$ представляет интервал булевых векторов $N(\alpha)$, такой что

$$N(\alpha) = \{\gamma \in E_n^2 : \gamma \leqslant \alpha\}.$$

Назовем *интервальным расширением полностью определенной булевой функции* $f(x_1, \ldots, x_n)$ такую полностью определенную троичную функцию $f^\times(x_1, \ldots, x_n)$, что значение $f^\times(X)$ на троичном векторе $\alpha \in {E^\times}_n$ определяется следующим образом:

1) $f^\times(\alpha) = 1$, если для $\forall \gamma \in N(\alpha)$: $f(\gamma) = 1$ $(N(\alpha) \subseteq M_1^f)$;
2) $f^\times(\alpha) = 0$, если для $\forall \gamma \in N(\alpha)$: $f(\gamma) = 0$ $(N(\alpha) \subseteq M_0^f)$;
3) $f^\times(\alpha) = \times$, если $\exists \gamma_i \in N(\alpha)$: $f(\gamma_i) = 1$ и $\exists \gamma_j \in N(\alpha)$: $f(\gamma_j) = 0$ $(N(\alpha) \bigcap M_1^f \neq \varnothing$ и $N(\alpha) \bigcap M_0^f \neq \varnothing)$.

2. Метод троичного моделирования комбинационной схемы

В работе [4] задача точного троичного моделирования комбинационной схемы на троичном векторе рассматривается как задача вычисления значений интервальных расширений булевых функций, реализуемых выходами схемы.

В данной работе для простоты изложения будем говорить: «троичное значение булевой функции $f(X)$ на троичном векторе α», имея в виду троичное значение интервального расширения булевой функции $f(X) - f^\times(X)$, на троичном векторе α. Будем иногда говорить: «троичное моделирование булевой функции $f(X)$ на троичном векторе α», имея в виду вычисление троичного значения $f^\times(\alpha)$.

Далее опишем предлагаемый метод троичного моделирования булевой функции. Метод основан на представлении булевой функции в виде ROBDD графа. Для троичного моделирования комбинационной схемы необходимо представить функции, реализуемые выходами схемы, в виде ROBDD.

Введем бинарную операцию $\cup$ над троичными аргументами. $z = x \cup y$, где x и y — троичные значения, определим следующим образом:

1) если один из аргументов принимает значение $\times$, то $z = \times$,
2) $1 \cup 1 = 1$,
3) $0 \cup 0 = 0$,
4) $0 \cup 1 = \times$,
5) $1 \cup 0 = \times$.

Вычислим троичное значение булевой функции f, представленной ROBDD графом, на троичном векторе α, выполнив сокращенный обход графа. При таком обходе для вершин вычисляются троичные значения. Если для некоторой вершины троичное значение уже вычислено, то оно сохраняется и при необходимости используется при прохождении других путей. Перед выполнением обхода припишем каждой вершине ROBDD значение ε.

Метод вычисления троичного значения булевой функции f, представленной ROBDD, на троичном векторе α.

1. Для функции $f_w(X)$, представленной в виде ROBDD с корнем w, троичное значение $f_w(\alpha)$ вычисляется следующим образом: если w — терминальная вершина со значением 0 (1), то $f_w(\alpha) = 0(1)$, иначе:
   а) если $\alpha_x = 1$, то $f_w(\alpha) = f_{high(w)}(\alpha)$,
   б) если $\alpha_x = 0$, то $f_w(\alpha) = f_{low(w)}(\alpha)$,
   в) если $\alpha_x = \times$, то $f_w(\alpha) = f_{low(w)}(\alpha) \cup f_{high(w)}(\alpha)$,
   где x — переменная, соответствующая вершине w, $low(w)$ и $high(w)$ — потомки вершины w по значениям 0 и 1 переменной x, α_x — значение компоненты вектора α, соответствующей переменной x.
2. Обход графа прекращается когда выполняется хотя бы одно из условий:
   а) для некоторой вершины графа получено значение $\times$. Тогда $f(\alpha) = \times$. Или
   б) получено значение в корне ROBDD. Оно является значением $f(\alpha)$.

Значение $\times$, полученное для некоторой вершины графа в п. 2, *а*, означает, что при обходе встретилась хотя бы одна ветвь, ведущая в терминальную вершину 1, и хотя бы одна ветвь, ведущая в терминальную вершину 0, в этом случае $f(\alpha) = \times$.

Заметим, что, если $f(\alpha)$ равно 1 или 0, то для определения значения $f(\alpha)$ необходимо выполнить сокращенный обход без предварительного выхода по п. 2, *а*.

В работе [5] представлен метод троичного моделирования комбинационных схем на основе вероятностного моделирования, выполняемого по ROBDD графу. Предлагаемый в данной работе метод требует меньших вычислительных затрат по сравнению с вероятностным подходом, так как выполняется сокращенный обход графа и не требуется перемножение вещественных чисел.

Список литературы

[1] *Голубева О. И.* Троичное моделирование синхронных последовательностных схем, основанное на ROBDD представлении функций // Сборник статей XVI Междунар. науч.-практ. конф. «Кулагинские чтения: техника и технологии производственных процессов». — Ч. 2. — Чита: ЗабГУ, 2016. — С. 131–136.

[2] *Голубева О. И.* Троичное моделирование синхронных схем с памятью на последовательностях произвольной длины, основанное на ROBDD представлении функций // Таврический научный обозреватель. — 2017. — № 3(20). — С. 129–140.

[3] *Голубева О. И.* Интервальные расширения булевых функций и троичное моделирование последовательностных схем // Таврический научный обозреватель. — 2017. — № 5(22). — С. 208–220.

[4] *Голубева О. И.* Разработка и исследование методов моделирования и оценки мер тестопригодности логических схем : дис. . . . канд. техн. наук: 05.13.01. — Томск, 2000.

[5] *Matrosova A., Golubeva O., Tsurikov S.* Random Approach to Correcting the Results of a Ternary Simulation // Compendium of papers. IEEE European Test Workshop. — Cagliari, Italy, 1997.

Модели и базы данных в контексте непрерывного образования

Грибанова—Подкина М. Ю.

m.gribanova-podkina@rambler.ru

Балашовский институт (филиал) СГУ имени Н. Г. Чернышевского, Россия

Рассматриваются вопросы, связанные с подготовкой специалистов в сфере разработки, проектирования и администрирования баз данных. Затронуты формы образования, которые в непрерывном цикле способны обеспечить подготовку таких кадров. Исследованы соответствующие профессиональные стандарты и сформулированы выводы по общетеоретическим и специальным знания и умениям, которыми должны обладать специалисты в сфере баз данных. Обоснована необходимость послевузовского профилирования в данной сфере.

Ключевые слова: базы данных, непрерывное образование, специалист в области баз данных

Введение

Информационные системы, обслуживающие практически все секторы экономики, достигли на современном этапе внушительного уровня развития как по своему архитектурному построению, так и по количеству обрабатываемой информации. Профессионалы в области разработки программных продуктов знают, насколько сложной стала структура современного промышленного программного обеспечения. Она включает в себя множество взаимосвязанных компонентов, часто распределенных не только логически, но и физически расположенных на значительных расстояниях. Разработка, да и обслуживание таких систем — это сложная задача, которая может быть решена только специалистами с высокой квалификацией. При этом сложность систем диктует необходимость наличия узких специалистов — профессионалов в своей определенной области, что, впрочем, не отменяет их представлений о функционировании системы как единого целого и умения использовать свои решения в контексте общей задачи. Таких специализаций в сфере IT великое множество: это системные и бизнес-аналитики, архитекторы, разработчики и администраторы баз данных, программисты (с различной специализацией), тестировщики (тоже по областям) и многие другие.

Понятно, что в таких условиях невозможно подготовить нужного специалиста в рамках одной образовательной программы: многообразие программных решений, скорость развития технологий и гибкость рынка IT слишком велики. Отчасти решить эту проблему может система подготовки кадров, построенная на непрерывном цикле форм образования, который включает в себя следующие составляющие:

1. вузовская подготовка специалистов;
2. индивидуальная подготовка по дистанционным курсам;
3. обучение по интенсивным программам учебных центров;
4. корпоративные программы обучения [1].

Следует отметить, что результаты подготовки по каждой форме образования сильно отличаются: для вузовской подготовки характерна классическая базовая подготовка; дистанционные курсы и программы учебных центров интересны в плане предложений по изучению конкретных учебных продуктов, а корпоративные программы нацелены на получение готового специалиста для конкретной компании, владеющего строго определенными технологиями. Естественно, ни одно звено в этой цепочке не является лишним.