Математические методы верификации схем и программ

mk.cs.msu.ru → Лекционные курсы
→ Математические методы верификации схем и программ

Блок 5

Дедуктивная верификация программ:
возможности автоматизации,
слабейшее предусловие

Лектор:
Подымов Владислав Васильевич
E-mail:
valdus@yandex.ru

ВМК МГУ, 2022/2023, осенний семестр

Насколько просто **автоматизировать** проверку корректности программ?

Иллюстрация, показывающая, насколько это непростая задача:
доказательство истинности триплета

$$\{\mathtt{x} > 0 \,\&\, \mathtt{y} > 0 \,\&\, \mathtt{z} > 0 \,\&\, \mathtt{n} > 2\}$$
$$\text{if } \mathtt{x}^{\mathtt{n}} + \mathtt{y}^{\mathtt{n}} = \mathtt{z}^{\mathtt{n}} \text{ then } \mathtt{u} := 0; \text{ else } \mathtt{u} := 1; \text{ fi}$$
$$\{\mathtt{u} > 0\}$$

в $\mathcal{I}_{ar}$ — это доказательство Великой теоремы Ферма

Более того, проблема корректности программ в общем случае и даже в ряде достаточно простых случаев **неразрешима**:

- программы с операцией $+$ и отношением $<$ в интерпретации $\mathcal{I}_{ar}$ полны по Тьюрингу
- любое нетривиальное семантическое свойство частично-рекурсивных функций, а значит, и машин Тьюринга, неразрешимо (*Rice. Classes of recursively enumerable sets and their decision problems. 1953*)

Слабейшее предусловие (weakest precondition) для программы π и постусловия ψ в интерпретации $\mathcal{I}$ — это формула $wpr(\pi, \psi, \mathcal{I})$, для которой выполнены следующие условия:

1. $\mathcal{I} \models \{wpr(\pi, \psi, \mathcal{I})\}\pi\{\psi\}$
2. Для любой формулы χ, такой что $\mathcal{I} \models \{\chi\}\pi\{\psi\}$, верно $\mathcal{I} \models \chi \to wpr(\pi, \psi, \mathcal{I})$

Теорема. $\mathcal{I} \models \{\varphi\}\pi\{\psi\} \quad \Leftrightarrow \quad \mathcal{I} \models \varphi \to wpr(\pi, \psi, \mathcal{I})$

Доказательство. Очевидно? (*следует из определений слабейшего предусловия и истинности триплета*)

Будем писать $wpr(\pi, \psi)$ вместо $wpr(\pi, \psi, \mathcal{I})$, если слабейшее предусловие не зависит существенно от интерпретации $\mathcal{I}$.

Для полной автоматизации проверки корректности программ достаточно иметь два алгоритма:

1. Алгоритм проверки истинности формул логики предикатов
   - Этот алгоритм зависит от выбора интерпретации программ и в общем случае может и не существовать
2. Алгоритм вычисления слабейшего предусловия для произвольных программ и постусловий
   - С этим алгоритмом дела обстоят немного лучше:

Теорема.

- $wpr(\mathtt{x} := t;, \psi) = \psi\{\mathtt{x}/t\}$
- $wpr(\pi_1\ \pi_2, \psi) = wpr(\pi_1, wpr(\pi_2, \psi))$
- $wpr(\text{if } C \text{ then } \pi_1 \text{ else } \pi_2 \text{ fi}, \psi) =$
$$C \,\&\, wpr(\pi_1, \psi) \vee \neg C \,\&\, wpr(\pi_2, \psi)$$

Доказательство. Самостоятельно

То есть проверка правильности императивных программ без циклов настолько же трудна, насколько и проверка истинности формул

$$R_{\text{while}}\colon \frac{\{\varphi\}\text{while } C \text{ do } \pi \text{ od}\{\varphi \,\&\, \neg C\}}{\{\varphi \,\&\, C\}\pi\{\varphi\}}$$

Для применения правила R_{while} (а также для аннотации цикла, и для вычисления слабейшего предусловия цикла) необходимо предварительно найти подходящую формулу φ —

инвариант цикла

Автоматическая генерация инвариантов циклов, как правило, является **основной проблемой** автоматизации проверки правильности программ

Эта проблема осложняется тем, что

1. никто не гарантирует существование свойства данных, выражаемого инвариантом цикла
2. если такое свойство существует, то никто не гарантирует, что это свойство можно записать в виде формулы в используемой сигнатуре