AZƏRBAYCAN RESPUBLİKASI MİLLİ ELMLƏR AKADEMİYASI
KİBERNETİKA İNSTİTUTU
İNFORMASİYA TEXNOLOGİYALARI İNSTİTUTU

«İnformasiyalaşdırma, Kibernetika və informasiya texnologiyalarının müasir problemləri» Respublika elmi konfransının

(Bakı, 28-30 aprel 2003 il)

ƏSƏRLƏRİ

I CİLD

İNFORMASİYA TEXNOLOGİYALARININ NƏZƏRİ VƏ TƏTBİQİ PROBLEMLƏRİ

NƏZARƏT VƏ İDARƏETMƏ SİSTEMLƏRİ

В работе проводены объемные вычислительные експерименты и получены численные результаты .

В этом эксперименте при различном количестве источников информации и длину сообщений определены значения задержи обслуживания в зависимости от интенсивности входного потока .

Потери информации в такой сети могут возрастать за счет появления потерь в терминалах .

С целью оценки характеристик сети конкретного назначения могут быть использованы полученные результаты в рамках однофазной модели системы массового обслуживания.

Следует отметить, что при более детальном изучении системы массового обслуживания такого вида целесообразно применять методы имитационного моделирования.

ЛИТЕРАТУРА

1. А.З.Меликов, Л.А.Пономаренко, Н.А. Рюмшин Математические модели многопотоковых систем обслуживания. –К.1991 – 264с.
2. А.А.Алиев, Б.Г.Исмайлов Анализ характеристик многопотоковых сетей обслуживания. “Радиоэлектроника, Информатика, Управления”.№2, 2001, с 66 – 69.
3. Ю.В.Новиков, С.В. Кондратенко Локальные сети. Архитектура, алгоритмы, проектирование, ЭКОМ, М.2001

УДК:681.3:16 Т.Г.КЯЗИМОВ, Т.Х.ФАТАЛИЕВ

О ФУНКЦИОНАЛЬНЫХ АНАЛОГАХ ЛОГИЧЕСКИХ ЧАСТЕЙ АЛГОРИТМОВ ВЫЧИСЛЕНИЯ В ПРОГРАММИРОВАНИИ

(ИИТ НАН АР)

В процессе программирования некоторых задач со сложными логическими выражениями и функциями, в силу ряда объективных причин, приходится использовать функциональные аналоги логических операций, выражений и функций . Данная ситуация может возникнуть и в случае, когда для решения задачи применяются программы оболочки (напр., Excel, Access и др.). Но, с другой стороны, известно, что выполнение алгоритмов с логическими разветвлениями в компьютерах требует намного больше времени, чем выполнение алгоритмов с математическими выражениями. Поэтому считаем, что замена логических операций, выражений и функций их функциональными аналогами в вычислительных алгоритмах может существенно повлиять на качество составляемых программ и тем самым повысить эффективность их выполнения.

Данная работа посвящена вопросам построения, исследования и применения функциональных аналогов логических операций, выражений и функций, ибо считаем, что в литературе некоторые из этих функций встречаются в теории вероятностей случайных величин [1]. Не вдаваясь в подробности , приведем функциональные аналоги простых логических операций [2], т.е. речь идет о таких действительных функциях $f(x, y)$, которые, аналогично соответствующим логическим операциям, принимали бы значения $f(x, y) = 0;1$, при значениях $x, y = 0;1$.

Пусть x и y числовые переменные принимающие значения $x,\ y = 0;1$, аналогично логическим значениям *"FALSE"*, *" TRUE"* соответственно. Тогда действительные функции

$$\hat{f}(x,\, y) = x \cdot y\,, \tag{1}$$

$$\overset{\neg}{f}(x,\, y) = 1 - x\,, \tag{2}$$

$$\overset{\vee}{f}(x,\, y) = 1 - (1-x)(1-y) = x + y - xy\,, \tag{3}$$

$$\overset{\Rightarrow}{f}(x,\, y) = 1 - x(1-y), \quad \overset{\Leftrightarrow}{f}(x,\, y) = 1 - x - y + 2x\,, \tag{4}$$

будут аналогами логического умножения, отрицания, сложения, импликации и эквиваленции соответственно.

Итак, используя вышеприведенные функции любые логические выражения можно перевести к функциональным и вычислить их числовые значения. Однако необходимо учитывать, что этими функциями можно воспользоваться только при значениях аргументов $x, y = 0;\ 1$.

Теперь, приведем функциональные аналоги соотношений типа

$$x = a, \quad x \neq a, \quad x > a, \quad x < a, \quad a < x < b,$$

где x, a, b - произвольные вещественные числа. Для этой цели воспользуемся свойствами функции

$$\operatorname{sgn}(x) = \begin{cases} 1, & x > 0; \\ -1, & x < 0; \\ 0, & x = 0, \end{cases} \tag{5}$$

которая, как стандартная функция, используется почти во всех вычислительных программных средствах.

Не сложно увидеть, что функция

$$\overset{\neq}{f}(x,\, a) = |\operatorname{sgn}(x-a)| \tag{6}$$

соответствует соотношению $x \neq a$.Тогда функция соответствующая ее функциональному отрицанию

$$f(x,\, a) = 1 - |\operatorname{sgn}(x-a)| \tag{7}$$

будет удовлетворять требованиям соотношения $x = a$.

Воспользуясь уже известными свойствами функции (5), для $x > a$ и $x < a$ напишем функции

$$\overset{>}{f}(x,\, a) = 0{,}5\,(\operatorname{sgn}(x-a) + |\operatorname{sgn}(x-a)|), \tag{8}$$

$$\overset{<}{f}(x,\, a) = 0{,}5\,(\operatorname{sgn}(a-x) + |\operatorname{sgn}(a-x)|), \tag{9}$$

которые, соответственно могут быть использованы как их функциональные аналоги.

Произведением функций (8) и (9) можно воспользоваться как функциональным аналогом соотношения $a < x < b$:

$$\overset{<>}{f}(x,\, a,\, b) = \overset{<}{f}(x,\, a) * \overset{>}{f}(x,\, a) \tag{10}$$

Используя функции (6) - (10) при составлении программ содержащие вычисления значений разветвляющих функций, можно избежать логических разветвлений алгоритмов. Покажем это на одном простом примере. Пусть необходимо составить программу (алгоритм) для вычисления значения действительной функции

$$f(x)=\begin{cases} f_1(x), & x \in (a,b); \\ f_2(x), & x \overline{\in} (a,b). \end{cases} \tag{11}$$

Если вместо функции (11) воспользоваться функцией

$$\tilde{f}(x)=p\,f_1(x)+(1-p)f_2(x),$$

где $p=\overset{<>}{f}(x,a,b)$, которая совпадает с ней, то увидем, что при составлении программы, необходимость использования логического разветвления алгоритма отпадет.

Подобный прием можно применять к любым многоразветвляющимся функциям.

Приведем конкретный практический пример проиллюстрирующий данный подход. Пусть необходимо составить программу для вычисления подоходного налога от заработной платы работника по следующей прогрессивной шкале:

Сумма месячного дохода, подлежащая к налогообложению	Сумма налога
До 100 000 м.	Не подлежать к налогообложению
100,001 - 600 000 м.	12% от суммы прев. 100 000 м
600 001 - 1 400 000 м.	60 000 м. + 20% от суммы прев. 600 000 м.
1 400 001 - 3 000 000 м.	220 000 м. +25% от суммы прев. 1 400 000 м.
3 000 001 - 5 000 000 м.	620 000 м. +30% от суммы прев. 3 000 000 м.
Более 5 000 000 м.	1 220 000 м. +35% от суммы прев. 5 000 000 м.

Формула для вычисления будет выглядеть следующим образом:

$$\begin{aligned} P(x) = 0{,}5\{ & [\mathrm{sgn}(x-100001)-\mathrm{sgn}(x-600001)](x-100000)\cdot 0{,}12 + \\ & + [\mathrm{sgn}(x-600001)-\mathrm{sgn}(x-1400001)]((x-600000)0{,}2+60000) + \\ & + [\mathrm{sgn}(x-1400001)-\mathrm{sgn}(x-3000001)]\cdot((x-1400000)0{,}25+220000) + \\ & + [\mathrm{sgn}(x-3000001)-\mathrm{sgn}(x-5000001)]((x-3000000)0{,}3+620000) + \\ & + [\mathrm{sgn}(x-5000001)+|\mathrm{sgn}(x-5000001)|]((x-5000000)0{,}35+1220000)\}, \end{aligned}$$

где x - сумма заработной платы.

Нетрудно увидеть, что часть составленной программы по этой формуле будет содержать один оператор присваивания в место шести операторов условного перехода (напр.: при использовании программы Microsoft Excel логическая функция IF будет отсутствовать).

В заключении отметим, что предложенный метод обхода логических операций, выражений и функций может быть использован и как дополнительный инструмент в программировании.

ЛИТЕРАТУРА

1.Б.В. Гнеденко Курс теории вероятностей. Наука, М. –1988 г.

2. И.А.Лавров, Л.Л. Максимова Задачи по теории множеств, математической логике и теории алгоритмов. 4-е изд., Физматлит, М-2001 г.