Kesra NERMEND
UNIWERSYTET SZCZECIŃSKI, INSTYTUT INFORMATYKI W ZARZĄDZANIU

Sprzętowa implementacja reprezentacji „wartość średnia-wariancja”

Dr inż. Kesra NERMEND

Ukończył studia na Wydziale Techniki Morskiej Politechniki Szczecińskiej w 1997 roku. Doktorat obronił w roku 2003 na Wydziale Nauk Ekonomicznych i Zarządzania Uniwersytetu Szczecińskiego. Obecnie pracuje na stanowisku adiunkta w Instytucie Informatyki w Zarządzaniu WNEiZ US. Jego zainteresowania to analiza i klasyfikacja danych.

e-mail: kesra@szafir.univ.szczecin.pl

Streszczenie

W artykule przedstawiono reprezentację „wartość średnia-wariancja”. Pokazano przykładowy sposób wyprowadzania formuł w zwykłych systemach liczbowych ze wzorów zapisanych w reprezentacji wartość średnia-wariancja oraz porównano dwa sposoby implementacji sprzętowej omawianej reprezentacji.

Słowa kluczowe: wartość średnia-wariancja, przeciążenie operatorów.

Hardware implementation of representation of value mean-variance

Abstract

In this paper the representation of mean-variance has been presented. Exemplary way of bringing out formulas in common numerical systems from expressions of representation of value mean-variance has been demonstrated. Two ways of hardware implementation of representation of value mean-variance have been compared.

Keywords: value mean-variance, overriding operators.

1. Wprowadzenie

Liczby wykorzystywane w obliczeniach możemy pogrupować w pięć zbiorów: zbiór liczb naturalnych, całkowitych, wymiernych, rzeczywistych oraz zespolonych. Przy czym każdy z tych zbiorów jest rozszerzeniem poprzedniego. Najszerszym z nich jest zbiór liczb zespolonych zawierający wszystkie liczby naturalne, całkowite, wymierne, rzeczywiste oraz liczby urojone nie występujące w pozostałych zbiorach.

Wykonując obliczenia z użyciem jakichkolwiek z liczb należących do tych zbiorów zakładamy, że wartości, które te liczby reprezentują są dokładne. Jednak w istocie tak nie jest. Aby móc wykonać obliczenia musimy najpierw przeprowadzić estymację punktową, sprowadzając naszą niepewność pomiarową do jednej wartości. Naturalnym rozszerzeniem wszystkich tych zbiorów liczb byłby zatem zbiór liczb zawierających więcej niż jeden estymowany parametr. Przykładem tego rodzaju zbioru liczb jest zbiór liczb w reprezentacji „wartość średnia-wariancja”. W reprezentacji tej mamy dwa estymowane parametry: wartość średnią i wariancję. Ze względu na to, że zbiór ten zawiera on dwa parametry będące liczbami dowolnego typu, może on być rozszerzeniem liczb naturalnych, całkowitych, wymiernych rzeczywistych, jak i zespolonych. Reprezentacja ta ma szerokie zastosowanie w analizach ekonomicznych między innymi do przygotowania danych do grupowania danych i klasyfikacji [1-5].

Implementując algorytmy zawierające liczby w reprezentacji „wartość średnia-wariancja”, możemy postępować dwojako. Najprostszym i dającym najbardziej czytelny kod sposobem jest przeciążenie operatorów dodawania, odejmowania, mnożenia i dzielenia. Możemy również na podstawie wzorów w reprezentacji splotowej wyprowadzić odpowiednie wzory dla zwykłych liczb. W artykule zostanie przedstawione porównanie implementacji sprzętowej obu sposobów przy zastosowaniu do kilku wybranych przykładów.

2. Reprezentacja wartość średnia-wariancja

Każdy zbiór pewnych wartości możemy bardzo dobrze scharakteryzować za pomocą rozkładu. Wielokrotne łączenie zbiorów o różnych rozkładach, w myśl centralnego twierdzenia granicznego, prowadzi do powstania zbioru, którego rozkład coraz bardziej przypomina rozkład normalny, stąd przy dużych zbiorach danych z reguły mamy do czynienia z tym rozkładem. Rozkład normalny charakteryzują dwa parametry – wartość średnia i wariancja. Posiadając oba te parametry możemy wyznaczyć odpowiadający im rozkład normalny.

W referacie [6] zaproponowano system liczbowy dla obliczeń na rozkładach oraz jego reprezentację uproszczoną „wartość średnia-wariancja” dla rozkładu normalnego. W systemie tym dodawanie i mnożenie zdefiniowano z wykorzystaniem aksjomatów ciała. Pozostałe dwie operacje – odejmowanie i dzielenie zdefiniowano na podstawie dwóch poprzednich z wykorzystaniem elementu przeciwnego i odwrotnego.

Ten system liczbowy dzięki temu, że spełnia wszystkie aksjomaty grupy abelowej, może zastąpić zwykłe systemy liczbowe w przypadku wielu prostych metod. Dzięki temu, oprócz informacji o wartości średniej, uzyskujemy również informację o wariancji otrzymanego wyniku.

3. Obliczanie parametrów prostej

Niech mamy dany wzór na prostą [7]:

$$y = ax + b. \tag{1}$$

Parametry a i b dla prostej przechodzącej przez punkty $\mathrm{p}_1(x_1, x_2)$ i $\mathrm{p}_2(x_1', x_2')$ możemy wyliczyć ze wzorów [7]:

$$a = \frac{x_2' - x_2}{x_1' - x_1}, \tag{2}$$

oraz [7]

$$b = x_2 - \frac{x_2' - x_2}{x_1' - x_1} x_1. \tag{3}$$

W reprezentacji wartość średnia-wariancja wzór na parametr a przyjmie postać [2]:

$$a = \left(\bar{x}_a; \sigma_a^2\right) = \frac{\left(\bar{x}_2'; \sigma_2^{2'}\right) - \left(\bar{x}_2; \sigma_2^2\right)}{\left(\bar{x}_1'; \sigma_1^{2'}\right) - \left(\bar{x}_1; \sigma_1^2\right)} = \left(\frac{\bar{x}_2' - \bar{x}_2}{\bar{x}_1' - \bar{x}_1}; -\frac{\left[\bar{x}_2' - \bar{x}_2\right]\left[\sigma_1^{2'} - \sigma_1^2\right]}{\left[\bar{x}_1' - \bar{x}_1\right]^2} + \frac{\sigma_2^{2'} - \sigma_2^2}{\bar{x}_1' - \bar{x}_1}\right) \tag{4}$$

Natomiast parametr b możemy zdefiniować następująco:

$$b = \left(\bar{x}_2; \sigma_2^2\right) - \left(\bar{x}_a; \sigma_a^2\right)\left(\bar{x}_1; \sigma_1^2\right) = \left(\bar{x}_2 - \bar{x}_1\bar{x}_a; \sigma_2^2 - \bar{x}_1\sigma_a^2 - \bar{x}_a\sigma_1^2\right). \tag{5}$$

Wykorzystując przeciążenie operatorów powyższe wzory możemy zapisać:

```
a = (x2p - x2)/(x1p - x1);
b = x2 - a*x1;
```

Natomiast bez przeciążenia operatorów:

```
tmp = 1/(x1_sr - x1p_sr);
a_sr = (x2_sr-x2p_sr)* tmp;
```

```
a_w =  (x2_w-x2p_w)* tmp - (x2_sr-x2p_sr)
*(x2_w-x2p_w)* tmp * tmp;
b_sr = x2_sr - a_sr*x1_sr;
b_w = x2_w - a_sr*x1_w - x1_sr*a_w;
```

W tym przypadku różnice w liczbie elementów są czterokrotne dla bramek NAND oraz 10-20% dla przerzutników Flip-flop na korzyść programu bez przeciążenia operatorów. Jedynie liczba ALU jest ponad dwukrotnie mniejsza w przypadku programu z przeciążeniem operatorów.

W prognozowaniu jako współrzędne x_1 i x_1' najczęściej przyjmuje się czas. W tym przypadku możemy założyć, że wariancja dla wszystkich chwil czasu jest taka sama. Wzory (4) i (5) ulegną wówczas uproszczeniu i przyjmą postać [2]:

$$a = \left(\overline{x}_a;\sigma_a^2\right) = \left(\frac{\overline{x}_2' - \overline{x}_2}{\overline{t}' - \overline{t}};\frac{\sigma_2^{2'} - \sigma_2^2}{\overline{t}' - \overline{t}}\right), \tag{6}$$

Wzór na wyliczenie współczynnika *b* będzie taki sam jak (5).

Powyższe wzory możemy zapisać w postaci programu korzystając z przeciążenia operatorów:

```
a = (x2p - x2);
tmp = 1/(x1p-x1);
a.set_sr(a.get_sr()*tmp);
a.set_w(a.get_w()*tmp);
ap.set_sr(a.get_sr()*x1);
ap.set_w(a.get_w()*x1);
b = x2 - ap;
```

Natomiast program bez przeciążenia operatorów będzie wyglądał następująco:

```
tmp = 1/(x1_sr - x1p_sr);
a_sr = (x2_sr-x2p_sr)*tmp;
a_w =  (x2_w-x2p_w)*tmp;
b_sr = x2_sr - a_sr*x1_sr;
b_w = x2_w - x1_sr*a_w;
```

W tym przypadku ze względu na ograniczenie przeciążenia operatorów do dodawania i odejmowania różnice w liczbie elementów nie przekraczają 0,5% na korzyść programu bez przeciążenia operatorów.

4. Podsumowanie

Dla każdego z zaprezentowanych wzorów wyjściowych napisano dwa warianty programów. Jeden z zastosowaniem przeciążenia operatorów, drugi bez przeciążenia, z wykorzystaniem wyprowadzonych formuł ze wzorów wyjściowych. Wyniki przedstawiono w tabeli 1.

Tab. 1. Porównanie liczby elementów niezbędnych dla wykonania implementacji wzorów

Tab. 1. The Comparison of number of elements required for formulas' implementation

Lp.	Wzór	Przy przeciążeniu operatorów			Bez przeciążenia operatorów		
		bramki NAND	przerzutniki Flip flop	ALUs	bramki NAND	przerzutniki Flip flop	ALUs
1	Wartość średnia	1528	64	0	780	34	0
2	Wariancja	2766	63	2	2766	63	2
3	Współczynniki a i b	21370	164	5	5760	148	8
4	Współczynnik a i b bez wariancji czasu	5430	132	4	5400	132	4

Najmniejszą różnicę uzyskano dla wariancji. W tym przypadku w obu wariantach kompilator w procesie optymalizacji uzyskał identyczny wynik liczby elementów. W pozostałych przypadkach kompilator dużo lepiej radził sobie z optymalizacją dla formuł zapisanych w zwykłym systemie liczbowym. Różnice w liczbie bramek NAND są dwukrotne i czterokrotne, natomiast różnice w liczbie przerzutników są dużo mniejsze, ale również wyraźne. Jedynie z optymalizacją liczby ALU kompilator radził sobie lepiej przy przeciążaniu operatorów. Programy zaprezentowane w artykule zostały wykonane dla układu Xilinx Spartan 3.

5. Literatura

[1] K. Nermend, M. Borawski: Using average-variance number system in calculation of a synthetic development measure, Polish Journal of Environmetal Studies, vol. 15, no. 4c, SMI 2006

[2] K. Nermend: Using average-variance representation in economic analyses, Polish Journal of Environmetal Studies, vol. 15, no. 4c, SMI 2006

[3] K. Nermend: A synthetic measure of sea environment pollution, Polish Journal of Environmetal Studies, Gdynia-Poland 2005

[4] K. Nermend: Application of artificial neural networks in the prediction of the Zachodniopomorskie voivodship's communes' incomes, Systemy Wspomagania Organizacji, SWO 2006, Ustroń 2006

[5] K. Nermend: Wykorzystanie reprezentacji średnia-wariancja do wspomagania klasyfikacji gmin, Nowe Technologie w Kształceniu na Odległość, Koszalin-Osieki 2006

[6] M. Borawski: Arytmetyka zbiorów pomiarów, Metody informatyki stosowanej w technice i technologii, Roczniki informatyki stosowanej Wydziału Informatyki Politechniki Szczecińskiej, Szczecin 2005

[7] I. N. Bronsztejn, K. A. Siemiendiajew, G. Musiol, H. Muhlig: Nowoczesne kompendium matematyki, PWN 2004

Artykuł recenzowany